

CA-7[®]

Systems Programmer Guide 3.3

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

THIS DOCUMENTATION MAY NOT BE COPIED, TRANSFERRED, REPRODUCED, DISCLOSED, OR DUPLICATED, IN WHOLE OR IN PART, WITHOUT THE PRIOR WRITTEN CONSENT OF CA. THIS DOCUMENTATION IS PROPRIETARY INFORMATION OF CA AND PROTECTED BY THE COPYRIGHT LAWS OF THE UNITED STATES AND INTERNATIONAL TREATIES.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

THE USE OF ANY PRODUCT REFERENCED IN THIS DOCUMENTATION AND THIS DOCUMENTATION IS GOVERNED BY THE END USER'S APPLICABLE LICENSE AGREEMENT.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227.7013(c)(1)(ii) or applicable successor provisions.

First Edition, September 2000

© 1988-2000 Computer Associates International, Inc.
One Computer Associates Plaza, Islandia, NY 11749
All rights reserved.

All trademarks, trade names, service marks, or logos referenced herein belong to their respective companies.

Contents

Chapter 1. Introduction	1-1
1.1 Summary of Revisions	1-2
1.1.1 Product Changes	1-2
1.1.2 Documentation Changes	1-5
1.2 CA-7 Overview	1-6
1.3 Functional Overview	1-7
1.3.1 Online Assistance	1-7
1.3.1.1 HELP Facility	1-7
1.3.1.2 PF Key Assignment	1-7
1.3.2 Workload Scheduling	1-8
1.3.3 Workload Sequencing	1-8
1.3.4 Work Flow Control	1-8
1.3.5 Virtual Resource Management	1-8
1.3.6 Automated Recovery Facility	1-9
1.3.7 Job Restart	1-10
1.3.8 Online Utility Execution	1-10
1.3.9 Security	1-10
1.3.10 JCL Overrides	1-11
1.3.11 External Communications Facilities	1-11
1.3.12 Batch Card Load Program	1-11
1.3.13 Workload Forecasting	1-11
1.3.14 Workload Planning	1-12
1.3.15 Workload Balancing	1-12
1.3.16 Workload Documentation	1-12
1.3.17 Management Level Reporting	1-13
1.3.17.1 Automated Performance Analysis (APA) Reporting	1-13
1.3.17.2 History Reporting	1-13
1.3.17.3 CA-Earl Reporting	1-13
1.3.17.4 CA-Easytrieve Plus Reporting	1-13
1.3.18 CA-7 Text Editor	1-13
Chapter 2. System Operations	2-1
2.1 System Structure	2-2
2.1.1 Central Control System	2-4
2.1.1.1 Application Programs	2-5
2.1.1.2 Base Calendars	2-5
2.1.2 ICOM	2-6
2.1.3 SVC and SMF Exits	2-6
2.1.4 CA-7 NCF	2-7
2.2 Database	2-8
2.2.1 Job Data Set	2-8
2.2.2 Dataset Data Set	2-8
2.2.3 Index Data Set	2-8
2.2.4 VRM Data Set	2-8
2.2.5 ARF Data Set	2-9
2.2.6 JCL Library(s)	2-10

2.2.7 Database Verification	2-10
2.2.8 Backup/Reload	2-10
2.3 Queues	2-11
2.3.1 Status Queues	2-11
2.3.2 Work Queues	2-11
2.3.3 Work Flow Through the Queues	2-12
2.4 Other Data Sets	2-16
2.5 Terminal Communications	2-18
2.5.1 Logical Terminals	2-18
2.5.2 Online Access	2-18
2.5.3 Batch Access	2-19
2.5.3.1 Batch Terminal Interface	2-19
2.5.3.2 U7SVC Facility	2-19
2.5.3.3 Trailer Step	2-19
2.5.3.4 Batch Card Load Program (BCLP)	2-19
2.5.3.5 Internal Terminals	2-20
2.5.3.6 CCI Terminals	2-20
2.5.4 BSAM Terminal (Browse Data Set)	2-20
Chapter 3. Installation Requirements	3-1
3.1 Operating Systems and Job Entry Systems Supported	3-2
3.1.1 Features Required	3-2
3.1.2 Memory Requirements	3-2
3.2 Data Set Requirements	3-3
3.2.1.1 Target Libraries for CA-7	3-5
3.2.2 Database Space Requirements	3-5
3.2.3 Database Files	3-6
3.2.3.1 Job Data Set	3-6
3.2.3.2 Dataset Data Set	3-7
3.2.3.3 Index Data Set	3-7
3.2.3.4 VRM Data Set	3-8
3.2.3.5 ARF Data Set	3-8
3.2.3.6 JCL Data Sets	3-9
3.2.4 Queue Files	3-10
3.2.4.1 Status Queues	3-10
3.2.4.2 Work Queues	3-11
3.2.4.3 Organization	3-11
3.2.4.4 Space Requirements	3-12
3.2.4.5 DASD Restrictions	3-13
3.2.5 Other Online File Requirements	3-13
3.2.5.1 Load Library	3-13
3.2.5.2 Checkpoint Data Set	3-13
3.2.5.3 Communications Data Set	3-14
3.2.5.4 Submit Data Set(s)/Internal Reader(s)	3-16
3.2.5.5 Log Data Set(s)	3-17
3.2.5.6 Queue Dump Data Set	3-18
3.2.5.7 VRM Dump Data Set	3-19
3.2.5.8 Batch Terminal Data Set(s)	3-19
3.2.5.9 Statistics Data Set	3-20
3.2.5.10 Workload Planning Data Set(s)	3-20

3.2.5.11	Browse Data Set	3-21
3.2.5.12	HELP Library	3-21
3.2.5.13	Override Library	3-22
3.2.5.14	CA-JCLCheck Options Data Set	3-22
3.2.5.15	CA-Driver Procedure Library	3-22
3.2.5.16	CA-7 JCLLIB Data Set	3-23
3.2.5.17	Model CA-Netman Transaction File	3-23
3.2.5.18	CA-Netman API Deferred Request Data Set	3-23
3.2.5.19	CA-7 Calendar Partitioned Data Set	3-24
3.2.6	Support File Requirements	3-25
3.2.6.1	Log History/Archive Data Sets	3-25
3.2.6.2	CA-7 Cross-Platform Scheduling Profile	3-25
3.2.6.3	CA-7 Cross-Platform Scheduling Checkpoint	3-26
3.3	ICOM Coupling Facility Structure	3-27
3.3.1	ICOM Coupling Facility Structure Lists	3-27
3.3.2	ICOM Coupling Facility Structure Implementation	3-28
3.4	Calendar Definition and Structure	3-30
3.4.1	CALENDAR Macro	3-32
3.4.1.1	Syntax	3-33
3.4.2	Coding Notes	3-35
3.5	Notification of CA-7 Job Completions Using CAIENF	3-36
3.6	Notification of CA-7 Browse Messages Using CAIENF	3-37
3.7	Workload Balancing	3-38
3.7.1	Defining the Production Environment	3-39
3.7.1.1	Tape Drives	3-40
3.7.1.2	CPU Use	3-41
3.7.1.3	Initiators and Job Class Structure	3-41
3.7.1.4	Job Start Times	3-41
3.7.1.5	Threshold Priorities	3-42
3.7.2	Implementing WLB	3-42
3.7.3	Defining the Selection Parameters	3-44
3.7.4	Workload Balancing Questionnaire	3-45
3.7.4.1	Workload Balancing Questionnaire Directions	3-47
3.7.4.2	WLBPDEF	3-47
3.7.4.3	TAPE1/TAPE2	3-47
3.7.4.4	CPU	3-48
3.7.4.5	INITR	3-48
3.7.4.6	CLBARR	3-50
3.7.4.7	STARTIME	3-50
3.7.5	Defining the Types of Tape Drives	3-53
3.7.6	WLB Macros	3-55
3.7.7	Coding WLB Macros	3-56
3.7.7.1	WLBPDEF	3-57
3.7.7.2	TAPE1	3-58
3.7.7.3	TAPE2	3-60
3.7.7.4	CPU	3-62
3.7.7.5	INITR	3-63
3.7.7.6	CLBARR	3-65
3.7.7.7	STARTIME	3-66
3.7.7.8	WLBPEND	3-67

3.7.8	Generating the Modules	3-68
3.7.9	Scheduling WLB Modules	3-69
3.7.10	Workload Balancing Commands	3-69
3.8	Tracking External Tasks	3-70
3.8.1	Usage Notes	3-70
3.8.2	Defining External Tasks To Be Tracked	3-71
3.8.2.1	SASSEXTL Example	3-72
3.8.3	Defining the Model Queue Records	3-72
3.8.4	\$L2EXTT Macro	3-73
3.8.5	Defining the SASSEXTT Module	3-76
3.8.5.1	SASSEXTT Example	3-76
3.9	Tracking External Data Sets	3-77
3.9.1.1	Usage Notes	3-77
3.9.2	Defining External Data Sets To Be Tracked	3-78
3.9.3	\$L2XDSN Macro	3-79
3.9.4	Defining the SASSXDSN Module	3-80
3.9.4.1	SASSXDSN Example	3-80
3.10	UNIX System Services Interface	3-81
3.10.1	File Permissions	3-82
Chapter 4.	Installation and Maintenance Procedures	4-1
4.1	Installation Verification	4-2
4.1.1	Step 1: Execute CA-7 Online	4-3
4.1.2	Step 2: Execute CA-7 ICOM	4-3
4.1.3	Step 3: 'Demand' Installation Verification Job	4-4
4.1.4	Step 4: 'Demand' the Test Network	4-5
4.1.5	Step 5: Force CA-7 Log Dump	4-7
4.1.6	Step 6: Check CA-7 Report Writing Using CA-Earl	4-7
4.2	Test Copy of CA-7	4-8
4.2.1	Test Copy Implementation Steps	4-8
4.3	CA-7 Test Copy Limitations	4-13
Chapter 5.	Initialization	5-1
5.1	Syntax Rules	5-2
5.1.1	Initialization File Statements	5-2
5.1.1.1	Sample Statement Format	5-3
5.1.1.2	Sample Statement Continuation Formats	5-3
5.2	Initialization File Statements and Keywords	5-4
5.2.1	RESIDENT Statement	5-10
5.2.1.1	Syntax	5-10
5.2.2	CUST Statement	5-12
5.2.2.1	Syntax	5-12
5.2.3	NEWS Statement	5-12
5.2.3.1	Syntax	5-12
5.2.4	INIT Statement	5-13
5.2.4.1	Syntax	5-13
5.2.5	SVCNO Statement	5-17
5.2.5.1	Syntax	5-17
5.2.6	APPLCTN Statement	5-22
5.2.6.1	Syntax	5-22

5.2.7 FMTBLK Statement	5-24
5.2.7.1 Syntax	5-24
5.2.8 CALBLK Statement	5-26
5.2.8.1 Syntax	5-26
5.2.9 UCC7VTAM Statement	5-27
5.2.9.1 Syntax	5-27
5.2.10 GROUP Statement	5-28
5.2.10.1 Syntax	5-28
5.2.11 LINE Statement	5-31
5.2.11.1 Syntax	5-31
5.2.12 TERM Statement	5-33
5.2.12.1 Syntax	5-34
5.2.13 INIT2 Statement	5-39
5.2.13.1 Syntax	5-39
5.2.14 SECURITY Statement	5-39
5.2.15 STATIONS Statement	5-40
5.2.15.1 Syntax	5-40
5.2.16 STNCAL Statement	5-41
5.2.16.1 Syntax	5-41
5.2.17 DBASE Statement	5-43
5.2.17.1 Syntax	5-43
5.2.18 ALOG1 Statement	5-46
5.2.18.1 Syntax	5-46
5.2.19 ALOG2 Statement	5-46
5.2.20 RESTART Statement	5-47
5.2.20.1 Syntax	5-47
5.2.21 SCHEDULE Statement	5-50
5.2.21.1 Syntax	5-50
5.2.22 CPU Statement	5-54
5.2.22.1 Syntax	5-54
5.2.23 CALENDAR Statement	5-56
5.2.23.1 Syntax	5-56
5.2.24 JCL Statement	5-57
5.2.24.1 Syntax	5-57
5.2.25 JCLCHECK Statement	5-61
5.2.25.1 Syntax	5-61
5.2.26 FORMAT Statement	5-63
5.2.26.1 Syntax	5-63
5.2.26.2 Memory Requirements	5-68
5.2.27 DAIO Statement	5-69
5.2.27.1 Syntax	5-69
5.2.28 NETMAN Statement	5-71
5.2.28.1 Syntax	5-71
5.2.29 OPTIONS Statement	5-72
5.2.29.1 Syntax	5-72
5.2.30 VRMOPTS Statement	5-81
5.2.30.1 Syntax	5-81
5.2.31 PFTERM Statement	5-82
5.2.31.1 Syntax	5-82
5.2.31.2 Examples	5-82

5.2.32 PFnn and PAnn Statements	5-83
5.2.32.1 Syntax	5-83
5.2.32.2 Examples	5-83
5.2.33 CANCEL Statement	5-84
5.2.33.1 Syntax	5-84
5.2.34 END Statement	5-85
5.2.34.1 Syntax	5-85
5.3 Offline Verification of Initialization File	5-86
5.3.1 Batch Edit Execution	5-86
 Chapter 6. Execution	 6-1
6.1 CA-7 Execution	6-2
6.1.1 REGION Size	6-2
6.1.2 PARM Values	6-3
6.1.3 Online Execution	6-4
6.1.3.1 CA-7 DD Statements	6-6
6.1.4 Batch Execution	6-11
6.1.4.1 RESIDENT Statement	6-11
6.1.4.2 INIT Statement	6-11
6.1.4.3 GROUP Statement	6-11
6.1.4.4 LINE/TERM Statement(s)	6-11
6.1.4.5 SECURITY Statement	6-12
6.1.4.6 STATIONS Statement	6-12
6.1.4.7 ALOG1/ALOG2 Statements	6-12
6.1.4.8 SCHEDULE Statement	6-12
6.1.4.9 CPU Statement	6-12
6.1.4.10 FORMAT Statement	6-12
6.2 Initialization Considerations	6-15
6.2.1 Date and Time Validation Option	6-15
6.2.2 Log Usage	6-16
6.2.3 Batch Job vs. Started Task	6-16
6.3 Startup Procedures	6-17
6.3.1 Startup Options	6-18
6.3.1.1 WARM Start (Normal Restart)	6-18
6.3.1.2 ERST Start (Emergency Restart)	6-19
6.3.1.3 COLD Start (Clear Workload Restart)	6-20
6.3.1.4 MOVQ Start (Move Queues Restart)	6-21
6.3.1.5 FORM Start (Format Queues Restart)	6-22
6.3.1.6 DORM Start (Dormant CA-7)	6-22
6.3.1.7 EDIT Start (Offline Verification of Initialization File)	6-22
6.4 Dormant CA-7	6-23
6.4.1 Procedure	6-23
6.4.2 ENQ Monitoring	6-24
6.5 Shutdown Procedures	6-25
6.5.1 OS CANCEL Command	6-25
6.6 ICOM Execution	6-26
6.6.1 ICOM Initialization	6-26
6.6.2 ICOM PARM Values	6-27
6.6.3 General Usage Considerations for ICOM	6-29
6.6.4 ICOM WTOR/MODIFY Replies	6-30

6.6.5 ICOM Coupling Facility Exploitation	6-34
6.6.5.1 Coupling Facility Operating Modes for ICOM	6-34
6.6.6 Cross-Platform Tracking under ICOM	6-35
6.6.6.1 Cross-Platform Tracking ICOM PARM Values	6-35
6.6.6.2 Cross-Platform Tracking ICOM DD Statements	6-36
6.6.6.3 Cross-Platform Tracking ICOM WTOR/MODIFY Replies	6-37
6.7 CAIRIM Initialization Considerations	6-38
6.7.1 Overview	6-38
6.7.2 CAIRIM Execution	6-38
6.7.3 CA-7 Reinitialization Using CAIRIM	6-39
6.7.3.1 Normal Reinitialization	6-39
6.7.3.2 COLD Reinitialization	6-39
6.7.3.3 User Table Reload Reinitialization	6-40
6.7.4 Reinitialization Procedures	6-41
6.7.4.1 Normal Reinitialization Procedures	6-41
6.7.4.2 COLD Reinitialization Procedures	6-42
6.7.4.3 User Table Reload Reinitialization Procedures	6-42
6.8 Other CAIRIM Initialization Parameters	6-43
6.8.1 Specifying the Host NCF Node	6-43
6.8.2 Forcing the CA-7 SVC	6-44
6.8.3 Clearing the CA-7 CSA/ECSA Chains	6-44
Chapter 7. Backup and Recovery Considerations	7-1
7.1 Database	7-2
7.1.1 Backup Procedures	7-2
7.1.1.1 IDCAMS	7-3
7.1.1.2 CA-ASM2	7-4
7.1.1.3 DFDSS	7-5
7.1.1.4 SASSBK00	7-6
7.1.1.5 SASSBK00 DD Statements	7-7
7.1.1.6 SASSBK00 PARM Values	7-9
7.1.2 Reload Procedures	7-11
7.1.2.1 IDCAMS	7-11
7.1.2.2 CA-ASM2	7-12
7.1.2.3 DFDSS	7-13
7.1.2.4 SASSBK00	7-13
7.1.2.5 Database Reload JCL Considerations	7-14
7.1.3 DBPARMS Parameter Deck	7-15
7.1.4 UCC7DBASE Statements	7-16
7.1.4.1 Syntax	7-16
7.1.4.2 UCC7DBASE Data Types	7-18
7.1.4.3 IDCAMSDEFINE Parameters	7-19
7.1.5 Backup/Reload Reports	7-20
7.1.5.1 Fields	7-21
7.1.5.2 Fields	7-24
7.2 ARF Backup and Reload	7-25
7.2.1 ARF Backup Procedure - CA7ARK	7-25
7.2.1.1 CA7ARK JCL Procedure DD Statements	7-25
7.2.2 ARF Reload Procedure - CA7ARL	7-26
7.2.2.1 CA7ARL JCL Procedure DD Statements	7-26

7.2.3 ARF Reload Special Considerations	7-27
7.2.3.1 ARF Data Set IDCAMS Define Parameters	7-27
7.3 VRM Backup and Reload	7-28
7.3.1 VRM Backup Procedure	7-28
7.3.1.1 VRM Backup Procedure - CA7VBK	7-29
7.3.1.2 CA7VBK JCL Procedure DD Statements	7-30
7.3.1.3 SASSVDB0 Sample Report	7-30
7.3.2 VRM Reload Procedures	7-30
7.3.2.1 VRM Reload Procedure JCL	7-31
7.3.2.2 CA7VRL JCL Procedure DD Statements	7-31
7.3.3 VRM Reload Special Considerations	7-32
7.3.3.1 VRM Data Set IDCAMS Define Parameters	7-32
7.4 Terminals	7-33
7.4.1 VTAM Terminals	7-33
7.4.2 Batch Terminals	7-34
7.5 Workload Considerations	7-35
7.5.1 CA-7 Considerations	7-35
7.5.2 Other CA Products	7-35
7.5.3 Other Vendor Software Products	7-35
7.5.4 CA-7 Workload Documentation Considerations	7-36
7.5.5 CA-7 NCF Considerations	7-36
7.5.6 Magnetic Tape Considerations	7-36
7.5.7 CA-7 Recovery Aid Considerations	7-36
7.6 Recovery Aid	7-37
7.6.1 Recovery Aid Facility	7-37
7.6.1.1 Queue Milestones	7-37
7.6.1.2 Preserving the Log Data	7-38
7.6.1.3 Reports Generated	7-38
7.6.1.4 Requesting the Reports	7-39
7.6.1.5 Recovery Aid Procedures	7-40
Chapter 8. Log and History Data Set Management	8-1
8.1 Log Data Set Management	8-2
8.2 SASSHIS5	8-4
8.2.1 File Descriptions	8-6
8.2.2 Report Descriptions	8-7
8.3 SASSHIS6	8-9
8.3.1 File Descriptions	8-12
8.3.2 Report Descriptions	8-13
Chapter 9. User Exits and Modifications	9-1
9.1 CA-7 User Modifications Under SMP	9-2
9.2 Coding User Exits	9-3
9.2.1 Rules for Coding Online Exits	9-5
9.2.2 Register Descriptions for Online Exits	9-6
9.2.3 Rules for Coding Standard Exits	9-7
9.2.4 Testing User Exits	9-7
9.3 Online Exit Descriptions	9-8
9.3.1 Job Name Verification	9-8
9.3.2 JCL Submission	9-9

9.3.3	Utility Function Security Checking	9-11
9.3.4	ENQ/RESERVE	9-13
9.3.5	Queue Entry JCL	9-14
9.3.6	External DSN Access	9-15
9.3.7	Forecast Worksheet	9-17
9.3.8	Command Exit	9-18
9.3.9	Job Data Verification	9-19
9.3.10	JCL Attach Verification	9-21
9.3.11	Database Load Processing	9-21
9.3.12	SMF Feedback	9-22
9.3.13	Personal Scheduling Verification	9-24
9.3.14	CA-7 Console Terminal Output	9-26
9.3.15	SMF WTO Message	9-27
9.3.16	Browse Message	9-28
9.3.17	Logon/Password Verification	9-29
9.3.18	Logoff Exit	9-30
9.3.19	Dump Queue	9-31
9.3.20	Move Queue	9-32
9.4	Standard Exit Descriptions	9-33
9.4.1	Batch Card Load Program Exit	9-33
9.4.2	Database Load Processing	9-35
9.4.3	Problem Management Interface	9-37
9.4.4	Database Backup	9-38
9.4.5	Database Reload	9-40
9.4.6	Database Cross-Reference	9-40
9.4.6.1	Program-Job Cross Reference	9-41
9.4.6.2	DSNAME-Program Cross Reference	9-42
9.5	Changing Graph Definitions	9-43
9.5.1	Graph Table Definition	9-43
9.5.2	DEFGRPH Macro	9-44
9.5.2.1	Syntax	9-44
9.6	Adding Counters for New Graphs	9-47
9.6.1	JCL Attach Exit Example	9-47
9.6.2	Defining Single Graphs	9-48
9.6.2.1	Examples	9-48
9.6.3	Defining Comparison Graphs	9-48
9.6.3.1	Examples	9-48
9.7	Other System Modification Techniques	9-49
9.7.1	Batch Interface Message Table	9-49
9.7.2	Generic Unit Name/Device Type Table	9-51
9.7.3	Message Level Suppression	9-52
9.7.3.1	Examples	9-52
9.7.4	CA-Driver Customization	9-53
9.7.5	Reserved DDname Table - SASSPMDD	9-53
9.7.6	SMF Type 14/15 Record Exclusion/Inclusion Table	9-54
9.7.6.1	Usage Notes	9-54
9.7.6.2	\$L2XU83 Macro	9-54
9.7.6.3	Pattern Masking	9-55
9.7.6.4	Defining the SASSXU83 Module	9-56
9.8	Function Aliases for Formatted Screens	9-57

Chapter 10. Performance and Tuning	10-1
10.1 Monitoring CA-7 Performance	10-2
10.1.1 Automated Performance Analysis (APA)	10-2
10.1.2 CA-Earl Reports	10-2
10.1.3 History Reporting	10-3
10.1.4 /DISPLAY Command	10-3
10.2 Initialization File Considerations	10-4
10.2.1 Schedule Scan Parameters	10-4
10.2.2 SMF Processing	10-4
10.2.3 Queue IOB Count	10-5
10.2.4 Application Pool Sizes	10-5
10.2.5 Terminal Dispatching Priority	10-6
10.2.6 Memory-Resident Queues	10-7
10.3 Calendar Schedules	10-8
10.4 Operating System Considerations	10-9
10.4.1 Nonswappable	10-9
10.4.2 Dispatching Priority	10-9
10.5 Data Set Placement Recommendations	10-10
10.5.1 Queues on a Single Pack	10-10
10.5.2 Queues on Multiple Packs	10-10
10.5.3 Queues on Fixed-Head Disk Drives	10-11
10.5.4 Data Set Placement Restrictions	10-11
10.5.4.1 Communications Data Set	10-12
10.5.4.2 Checkpoint Data Set	10-12
10.5.4.3 CA-7 Load Library	10-12
10.5.4.4 Placement of Queue Data Sets	10-12
10.6 Database Controls	10-14
 Chapter 11. Troubleshooting	 11-1
11.1 Diagnostic Procedures	11-2
11.1.1 Collecting Diagnostic Data	11-3
11.1.2 Interpreting Diagnostic Data	11-4
11.2 Accessing the Online Client Support System	11-5
11.2.1 Requirements for Using CA-TCC	11-5
11.2.2 CA-TCC Security	11-5
11.2.3 Accessing CA-TCC	11-6
11.2.4 Accessing the Technical Support Phone Services Directory	11-7
11.2.5 CA-TCC Hotline	11-7
11.3 CA-TLC: Total License Care	11-8
11.4 Calling Technical Support	11-9
11.5 Product Releases and Maintenance	11-10
11.6 Requesting Enhancements	11-11
 Appendix A. Migrations	 A-1
A.1 Migrating Queues to New Devices	A-2
A.1.1 Unloading the Queues	A-3
A.1.1.1 Virtual Resource Management Considerations	A-3
A.1.2 Reloading the Queues	A-4
A.1.2.1 Virtual Resource Management Considerations	A-4

Index	X-1
------------------------	------------

Chapter 1. Introduction

The *CA-7 Systems Programmer Guide* is intended for systems programmers responsible for planning, installation, and maintenance of CA-7 software. It includes information about system structure, installation requirements and procedures, initialization, execution, maintenance, backup and recovery, user exits, user modifications, performance and tuning.

1.1 Summary of Revisions

This topic explains changes to both CA-7 and to the documentation.

1.1.1 Product Changes

CA-7 Version 3.3 contains the following major enhancements:

- Parallel Sysplex Exploitation

CA-7 can optionally maintain a memory structure in the Coupling Facility in which participating ICOMs record tracking data. One or more Host ICOM(s) read from the memory structure and write to the Communication data set. This can significantly reduce I/O contention and increase feedback throughput.

- UNIX System Services Interface

The OS/390 UNIX System Services (USS) CA-7 interface allows communication with CA-7 from the USS environment. The interface can be called directly from the UNIX shell or from the IBM USS batch interface (BPXBATCH).

- CA-7 CCI Interface

The CA-7 CCI interface allows two-way communication with CA-7 from other address spaces and environments. The interface can be engaged in a batch mode, in a REXX address environment or it can be called directly from a user program. It accepts single or stacked commands as input and returns the CA-7 output from the commands as if they had been executed in batch mode.

- Critical Path Monitoring

Through integration with CA-OPS/MVS II, Unicenter TNG and Unicenter TNG MVS Event Manager Option (MEMO), CA-7 can support the definition and monitoring of critical job flows within the CA-7 workload. CA-OPS/MVS II provides management and administration of critical path displays.

- Mixed Case Support in CA-7 Editor

Character translation controls can be set in the CA-7 Editor. New Editor subcommands 'UPPER' and 'MIXED' determine whether editor data is translated to uppercase or left "as is."

These subcommands are enabled with a new initialization file option. If this option is not coded, then all edit data is translated to uppercase.

- Job Completion Tracking Precision

CA-7 records job completion times in hundredths of seconds. This allows job completions to be discriminated with a high degree of precision, thus reducing the likelihood of requirement posting ambiguities where jobs complete within the same minute.

- Display Duplicate Days for RESOLVE

CA-7 can optionally display the duplicate RESOLVE day(s) in new message SRC1-137. This occurs when a job is scheduled to execute the same day under two or more different Schedule IDs. With this information one can more quickly and efficiently determine the source of the scheduling conflict.

- VRM Device Control

Virtual Resource Management (VRM) Device Control provides an alternative to Workload Balancing control of job submission based on tape drive availability. VRM resource count resources representing the number and type of storage devices used by the job are defined dynamically during CA-7 LOAD processing.

Workload Balancing only permits two types of tape drives. With VRM Device Control, the number and structure of device groups is determined by the user.

- CA-7 Command Retrieval

Command line input for CA-7 VTAM terminals is recorded in storage and may be retrieved with the /FETCH command. When the /PFnn command is used to associate /FETCH with a PF key, the CA-7 user can conveniently retrieve the last five CA-7 commands entered at an online terminal.

- CA-7 Base Calendar Security

CA-7 security can allow clients to define CA-7 base calendar names to an external security product and secure user access to individual base calendars.

- REXX Address Environment

Using the new CA-7 CCI interface, CA-7 allows REXX programs to pass commands to CA-7 and take action based on the output from those commands.

- Job 'Purge' Function

The DB.1 (Job) panel provides a new function, PURGE, which deletes all CA-7 database records related to a job. In addition to the standard delete processes, the PURGE function deletes incoming trigger definitions, requirement successor definitions, and the CA-11 CMT member for the job.

- Suppress LATE Designation

Through an Initialization File option, the PROMPTS field on the DB.1 (Job) panel can be used to indicate certain jobs should never be marked as LATE on status displays. This means operations and production control staff will not be distracted when test or non-critical jobs do not complete on time.

- CSA Chains Above the 16M Line

CA-7 CSA SMF and Trailer chains now reside in extended CSA (above-the-line), thereby reducing utilization of this critical resource.

- Automated Recovery Facility (ARF) Enhancements

CA-7 can optionally add a LOGON parameter to the ARF TSO SEND command to cause messages to be retained until the user logs on to TSO. Also, support for ARF has been added to the Database Transportability facility.

- Prior Run Queue Expansion

The maximum size of the Prior Run Queue is now approximately twice as large as in prior releases.

- CA-7 JCLCheck Common Component

The CA-JCLCheck Common Component is provided in place of the CA-7 JCL syntax checker.

- Documentation Files on Tape

The current CA-7 documentation files are provided in IBM Book Manager and PDF format on the product tape.

- Other Enhancements:

- SMF Purge records may optionally be sent to a test copy of CA-7. This allows detection of pre-execution JCL Errors by the test copy.
- The Scratch and Disk Queue Table queues can be formatted during a CA-7 ERST start which facilitates use of VIO to improve performance.
- The LJOB command provides a new option, LIST=RQEXCP, that lists only those requirements with a SKIP or ONLY indication.
- The reverse forecast commands, FRJOB and FRQJOB, have a new option, LIST=HDRS. This will limit the display to only the target job and all 'header' jobs.
- Database Transportability now supports a new keyword, NODSNS, for SASSDT30 which prevents the generation of data set definitions.
- The LQ family of commands (LREQ, LRDY, LACT, and so forth) now support a Schedule ID filter, SCHID=.
- The LRLOG command has a new sequence option, SEQ=REV, which causes entries to be displayed in reverse date/time sequence (most recent first).
- The OPTIONS initialization file statement has a new keyword DPROCCOM= to enable comment statements in CA-Driver procedures.
- The OPTIONS initialization file statement has a new keyword EXTSCHID= to set a default schedule ID for externally tracked jobs that are not assigned a non-zero schedule ID from the SASSEXTT table.
- The CA-7 CAIRIM initialization module now accepts a new reinitialization parameter (REINIT=UTABS) to reload only user defined table modules.
- The /DISPLAY command has a new STATUS option (/DISPLAY,ST=CA7) to describe the current copy of CA-7 (VTAM application ID and so forth).

1.1.2 Documentation Changes

The documentation for CA-7 Version 3.3 differs from previous releases as follows:

- The documentation set has been engineered to take advantage of the latest technology for online viewing, keyword searching, book marking, and printing. The set consists of a hard copy *CA-7 Getting Started* guide and Version 3.3 of CA-7 for OS/390 documentation in both IBM BookManager and Adobe Acrobat Reader format on the tape.
- Unicenter TNG Framework for OS/390 is composed of the services formerly known as CA90s and Unicenter TNG Framework.
- Reading Syntax Diagrams in the *CA-7 Commands Guide* explains how to read the command syntax used in all guides.

Technical changes are identified by a revision bar (|) in the left margin. Revision bars are not used for editorial changes and new manuals.

1.2 CA-7 Overview

CA-7 is a comprehensive Automated Production Control system. It has the capability to address the broad range of activities traditionally considered the responsibility of computer operation's production control. CA-7 is an online, realtime, interactive system which automatically controls, schedules and initiates work according to time-driven and/or event-driven activities.

1.3 Functional Overview

CA-7 capabilities and facilities include, but are not limited to, the following topics.

1.3.1 Online Assistance

CA-7 incorporates facilities to assist in using the system to control the production environment, to operate the system, and to retrieve information from the system.

1.3.1.1 HELP Facility

The HELP facility is an online function available at any CA-7 terminal. Information describing CA-7 features, applications and commands can be rapidly retrieved and displayed using this feature. The HELP command accesses an online HELP library and displays a general discussion of the requested application or command. Information concerning commands includes a brief description of the command, its keywords, formats, and uses.

1.3.1.2 PF Key Assignment

Program function (PF) keys may be assigned to define a specific command for a predetermined function. This enables the terminal user to enter common functions with a single keystroke per function. The program access (PA) keys can be used in a similar manner.

1.3.2 Workload Scheduling

CA-7 can schedule all activities associated with the production workload. In general, these time-driven and event-driven facilities are used to schedule workstation activities (for example, data entry, production control, mail room) and CPU jobs. However, these same facilities can be used to schedule any definable activity, whether or not that activity is related to the production workload or to CPU job processing.

CA-7 also has the ability to track batch jobs and started tasks that are submitted outside of CA-7 and data sets created by such jobs. The tracking is selective based on tables that are coded to specify the job names and/or data set names which are to be tracked. This is only available for CPUs that share the Communication data set with CA-7.

1.3.3 Workload Sequencing

Workload sequencing refers to facilities which prevent jobs from being executed before input tasks are successfully completed and job dependencies (successful input data set creation, predecessor job or network completion) are satisfied.

1.3.4 Work Flow Control

Although CA-7 automatically schedules and invokes a defined workload, sometimes it is necessary to circumvent scheduled work flow for new priorities. With CA-7, unscheduled interruptions are handled online so that revised priorities can be addressed immediately. Schedules can be moved forward and backward. Jobs and activities can be held, rushed or canceled online without time-consuming rescheduling activity.

Online prompting reminds appropriate workstations when scheduled work is late or in danger of becoming late. Prompting promotes work flow by drawing attention to required action on activities which might otherwise be overlooked or delayed.

1.3.5 Virtual Resource Management

The Virtual Resource Management facility (VRM) provides a mechanism to control job submission based on job resource use. A job to resource connection defines job usage and disposition of the resource and thereby allows an additional level of production workload control. Resource use is managed internally by CA-7 during the job submission and job/step completion processes and may be used to serialize a production cycle or critical resource access.

1.3.6 Automated Recovery Facility

The Automated Recovery Facility (ARF) for CA-7 provides the capability to closely monitor exception conditions for production jobs and to schedule recovery actions to execute at or near the point of failure.

One or more ARF definitions may provide the criteria used to evaluate the exception conditions for a job. A named set of ARF definitions that provide the criteria used to determine the exception conditions for a job is referred to as an ARFSET. ARF determines exception conditions based on criteria in the ARFSET that is named on the DB.1 panel for the job.

ARF specifically tests for the exception conditions that are described in the ARFSET for the job. Each ARFSET may contain up to 20 distinct definitions. Each definition in an ARFSET includes a description of an exception condition along with a set of responses that are to be executed when the exception is detected.

ARFSET definitions may specify criteria that are precisely tailored to fit exact exception conditions for an individual job or may be coded to apply to a broad range of exception conditions for any job.

Types of exception conditions monitored by ARF include, but are not limited to: late conditions, step or job completion errors (including tests for condition code errors, system and user abends) and elapsed time checking during execution or at job completion.

ARF can respond to exception conditions by issuing CA-7 or MVS commands. ARF can also submit and track special recovery jobs that are not part of the "normal" schedule. Special messages may be routed to TSO users or to the MVS console as part of ARF response processing. ARF can cause a job to be restarted, canceled or "force completed" as part of processing a job completion exception.

ARF attempts to respond to exception conditions at or near the time they are detected. For example, ARF can issue CA-7 or MVS commands immediately at the point in time when ARF detects a step completion code exception.

1.3.7 Job Restart

Under CA-7, jobs which abnormally terminate are automatically suspended and notification is made to a designated workstation advising of a need to perform a job restart. Through online commands, all jobs waiting for restart may be listed with job restart information. Job restart information identifies the last step successfully executed, the abend code, the restartable steps and, optionally, specific user-defined special instructions. Abended jobs can be restarted online after restart cleanup is completed.

If CA-11 is available, both restart cleanup and job restart can be accomplished online on a single screen.

1.3.8 Online Utility Execution

The CA-7 utilities allow online execution of commonly used IEHPROGM-type functions. Several utility commands are supported including Data Set Allocation, allocate and catalog, rename, scratch, uncatalog, list PDS and list catalog.

Online execution of utilities reduces time otherwise required to perform these functions, and eliminates dependence on other methods such as batch jobs, TSO, CA-Roscoe, and so forth.

1.3.9 Security

CA-7 supports both internal and external security. External security packages, such as CA-ACF2 and CA-Top Secret, can be used to control access to CA-7, which terminals an operator can use, which commands an operator can issue, which data sets an operator can access, and the authority associated to jobs submitted by CA-7. For detailed information about using external security with CA-7, refer to the *CA-7 Security Guide*.

An internal security matrix can be used if an external security package is not available or desired. With the CA-7 security matrix, five levels of security are provided:

- Terminal/Operator
- Operator/Application
- Application/Command
- Command/Function
- User ID/External Data Set

Through the security matrix, personnel are allowed to access only those functions of the system as defined by installation personnel. To further define security, there are exits available which allow interfacing with external security packages.

1.3.10 JCL Overrides

The JCL Override facility can dynamically include or omit override statements based on current execution requirements for a given job. Both scheduled and unscheduled override requirements are supported. CA-7 provides tools to validate the syntax of these statements.

Additionally, the CA-Driver component facilitates automatic manipulation of JCL during job submission. Some of the features include:

- Date manipulation in JCL (or in-stream data)
- Conditional expansion of JCL
- Using variable parameters in JCL
- Nesting procedures

1.3.11 External Communications Facilities

CA-7 provides several programs which facilitate communications between CA-7 and users outside the CA-7 address space (for example, batch jobs and online sessions). Among the programs provided for this purpose are SASSTRLR and U7SVC. These facilities flexibly accommodate a wide variety of needs but are most commonly used to communicate information to CA-7 about the availability of processing requirements for CA-7 jobs.

1.3.12 Batch Card Load Program

The Batch Card Load Program (BCLP) loads card-image data into data sets which may be specified as input requirements for CA-7 jobs. BCLP permits data set creation, replacement and modification by way of data set request cards. BCLP is one way that UCC7, the Central Control System, can be notified of the input requirements that permit it to coordinate the availability of input data with the jobs dependent on that data.

1.3.13 Workload Forecasting

The Workload Forecast facility provides several important functions. It allows you to:

- Project the scheduled workload for a given period of time.
- Verify newly defined schedules.
- Produce a hardcopy checklist of scheduled tasks in a worksheet format.

Criteria available for selecting data to be forecasted provide great flexibility in tailoring a forecast to meet specific needs.

1.3.14 Workload Planning

Workload planning is a modeling and simulation tool. Using an online forecast command, you can easily extract any subset of the workload or the entire workload definition. Once this workload definition has been created, it can be simulated with or without changes. Processing objectives and configurations can be handled in the same manner. Workload planning then simulates the production processing of the representative workload within the model processing objectives and configuration. The simulator can include work on request (that is, nonscheduled jobs) and rerun or restart work. Simulations are performed in batch mode. The CA-7 normal production control and scheduling functions are not affected by the planning function, so online processing continues as normal throughout the batch planning process.

1.3.15 Workload Balancing

Workload balancing dynamically balances CPU work based on user-defined processing objectives.

Through this facility, workload balancing and performance management objectives are predefined to CA-7. These objectives can be modified online at any time to accommodate situations which are not anticipated. CA-7 automatically selects and submits a mix of jobs to the host system which best meets the defined objective.

1.3.16 Workload Documentation

CA-7 allows documentation of the workload at many levels, from general descriptions to specific instructions. Documentation is added to the CA-7 workload definition through the CA-7 text editor and is available for reference at CA-7 terminals.

Including this documentation in the CA-7 database is optional. The documentation capability is provided to assist the production control staff. CA-7 does not require documentation to properly control work.

1.3.17 Management Level Reporting

CA-7 includes several tools that provide information on a data center's productivity. Management level reporting is done on a global basis for jobs and networks from historical data retained by CA-7. There are three categories of historical reports provided.

1.3.17.1 Automated Performance Analysis (APA) Reporting

APA will provide reports on performance information based on history. Information can be reported either online or in batch mode, and can range from the current date and time to two years prior. These reports are generated in summary graph format.

1.3.17.2 History Reporting

As work is processed under CA-7, activities are recorded in a Log data set. This information may be used to generate a variety of reports through the CA-7 History Reporting facility (for example, Work Scheduled and Not Run Report, Actual Processing Activity Report).

1.3.17.3 CA-Earl Reporting

An abbreviated yet powerful version of CA-Earl is provided with Unicenter TNG Framework for OS/390. With CA-Earl, several statistical type history analysis reports are provided. Some database inventory type reports are also provided.

1.3.17.4 CA-Easytrieve Plus Reporting

For clients with the CA-Easytrieve Plus product installed, report definitions are provided to produce the same statistical analysis and database inventory reports with CA-Easytrieve Plus as can be produced with CA-Earl.

1.3.18 CA-7 Text Editor

The CA-7 text editor is an interactive facility for creating, modifying and managing card-image data. With the text editor, job streams (JCL) can be created and submitted for processing online. It is also the mechanism for adding documentation (prose) to the CA-7 database. The ISPF editor replaces the CA-7 text editor if CA-7 is accessed through the TSO/ISPF interface.

Chapter 2. System Operations

This chapter describes the major components and facilities of CA-7.

2.1 System Structure

A good understanding of the overall structure of CA-7 is a prerequisite for data center personnel. CA-7 includes the following items:

- Central Control System
- ICOM (Independent Communications Manager)
- SVC and SMF exits
- CA-7 NCF (Network Communications Facility)
- CA-7 database
- CA-7 queues
- Terminal communications

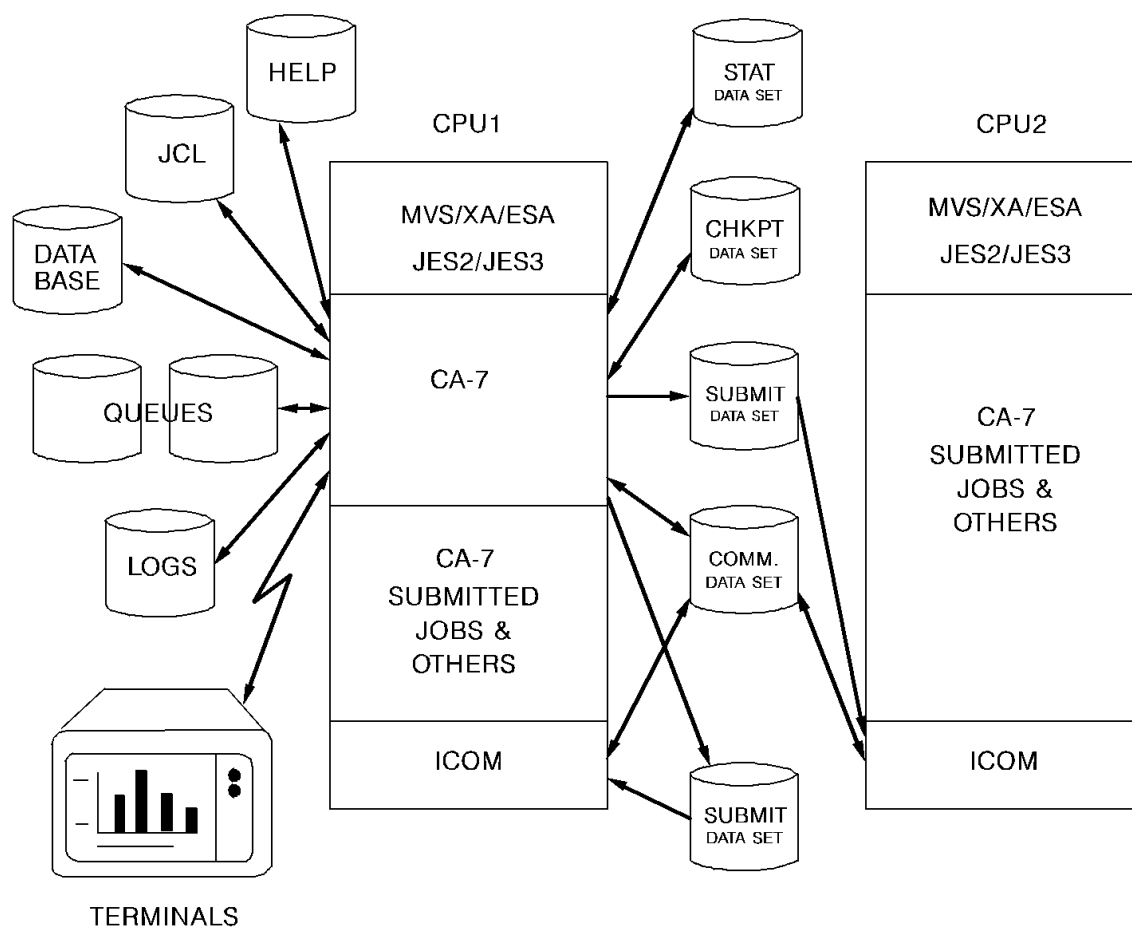


Figure 2-1. CA-7 Environment

Note: The submit data sets shown in the figure are optional. They are primarily used in a JES nonshared spool environment.

2.1.1 Central Control System

The Central Control System (Module UCC7) is the supervisor of the entire CA-7 system. All CA-7 facilities accessible to the user are under the control of the Central Control System. The Central Control System also controls execution of time- and/or event-driven functions, and all communications to and from terminals and between other CA-7 components.

The Central Control System may be executed as a started task or a batch job. When executed as a batch job, the Central Control System requires an initiator. The amount of virtual storage must be increased as the system grows. In multiple computer configurations, the Central Control System executes on only one computer, with an ICOM on each computer to which the Central Control System will be submitting jobs.

Central Control System functions are as follows:

- Queue data set access is centrally controlled so that system functions and the terminal operators can get to status information without undue delay. This function receives highest priority to make sure that the production workload is as current and as accurate as possible.
- Terminal network control handles the interface with VTAM to communicate with operators. This allows the various applications to operate independent of the type of teleprocessing network being used.
- The log management function controls writing of log records. Additionally, it monitors the usage of the log files. When a log file is full, this function swaps files and schedules a job to dump the full data set. These log dump jobs are monitored to verify success before the next swap occurs.
- Database access is handled so that applications can provide the services requested independent of the file configuration used.
- Subtask control provides an interface with the operating system for certain types of service. This approach allows the Central Control System to operate with minimal system overhead while supporting application requirements.
- SMF feedback is processed for work being tracked by the Central Control System. This data causes updating of status information in the queues and the database. By keeping status information current, the Central Control System can dynamically monitor the workload and react to changing requirements and conditions.
- Schedule scan is responsible for scheduling and controlling all CA-7 jobs and networks. It is activated automatically on a user-defined time interval or dynamically by internal events. Schedule scan functions include selecting work to be processed and placing it in the queues, performing an initial scan of requirements for newly scheduled work and prompting and reprompting for work that is late.
- Submit provides the JCL submission interface between the ready queue and the host job entry subsystem or submit data set. It is automatically activated when all requirements are satisfied for a job in the request queue.

2.1.1.1 Application Programs

CA-7 application programs provide the functions and activities required for effective use of automated production control (for example, prompting, schedule resolution, forecasting, and so forth). Each application provides multilevel capabilities for the activity it supports. The database maintenance application is an example. It provides the ability to update elements within a database entry as well as the ability to delete the entire entry or add a new entry.

Application modules are stored on the CA-7 load library. Source code for the applications is stored on the CA-7 source library. Some CA-7 applications provide user exits for the purpose of tailoring CA-7 to meet installation requirements. Information related to these exits is covered in Chapter 9, “User Exits and Modifications” on page 9-1.

2.1.1.2 Base Calendars

The base calendars built by an installation define the structure and processing year. These determine when scheduled processing can occur. The calendar criteria includes:

- Processing days available for work to be scheduled
- Holidays
- Beginning and ending days of each month (if not standard Gregorian)

The CALENDAR macro generates base calendars in batch mode. The user supplies macro parameters to define each calendar. A standard assemble and link-edit procedure is then used to build the calendar and store it in a library. Many different base calendars may be defined.

The CALMOD function (DB.2.8) can be used to define, modify, and delete base calendars in an online mode.

Detailed information on calendars can be found in 3.4, “Calendar Definition and Structure” on page 3-30.

2.1.2 ICOM

The Independent Communications Manager (ICOM) has direct contact with activity in the host system. It monitors the processing of computer job activity through data produced by the System Management Facility (SMF). ICOM monitors the SMF record types 15, and 30 (and optionally 14) created for jobs defined to, and submitted by, CA-7 and type 26 records created for all jobs. Optionally, the SMF type 4, 5, and 20 records can be used by CA-7 instead of record type 30.

The principal functions of ICOM are to:

1. Communicate SMF and trailer information to CA-7.
2. Submit jobs to the host system internal reader. (Optional)

ICOM is executed as either a started task or batch job and must be active whenever CA-7 is executed in production mode to perform automatic job submission, monitoring, and control.

In multiple computer configurations, ICOM must be active on each computer executing jobs to be controlled by CA-7. Additionally, in multiple computer configurations the CA-7 communications data set must be accessible to each computer on which ICOM executes. The communications data set is a transfer point between CA-7 and ICOM for SMF data, trailer data and control information.

ICOM continues to record activity associated with the CA-7 controlled work even if CA-7 is not active. This allows processing of data to be resumed by CA-7 when it is reactivated.

2.1.3 SVC and SMF Exits

The CA-7 SVC passes SMF information to ICOM collected by the CA-7 SMF interface processors, providing job and data set monitoring. The SVC is issued by ICOM and may be issued by each of the SMF interface processors.

Through the SMF processors, record types 15, 26, and 30 (and optionally 14) are monitored. Optionally, the type 4, 5, and 20 records may be used by CA-7 instead of record type 30. Only portions of these records are used. In no instance is an entire record processed or recorded. These records need not be written to the SMF data sets, but they must be generated.

2.1.4 CA-7 NCF

NCF, the Network Communications Facility, enables the user to process NJE work and have the work appear to CA-7 as if it were done locally. All the functions that CA-7 performs for jobs executed at remote nodes are the same functions that CA-7 performs for jobs executed locally.

Jobs that are scheduled by CA-7 can be routed by NJE to a remote node, and still be tracked by CA-7 using NCF. The SMF feedback for the jobs executed at a remote node returns to the originating node through the NCF/VTAM link. There are the usual event postings, dependencies, log records, and so forth, associated with a locally run job. Inside CA-7, various reports and screens show when and which jobs ran remotely.

JCL changes needed to cause a remote job to run correctly in the NCF environment can be made with the CA-7 DB.7 screen.

For detailed information on how NCF operates, consult the *CA-7 Interfaces Guide*.

2.2 Database

Information used to schedule and control work (jobs and workstations) is maintained in the CA-7 database. The database consists of two types of data. One type is the execution JCL for computer jobs. The other is the workload defined through and maintained by CA-7.

2.2.1 Job Data Set

The job data set contains information which reflects CPU job structure, activity relationships, execution characteristics and data set relationships for each CA-7 controlled job. Each job member defines the structure of the job and contains pointers to information in the dataset and index data sets. Jobs are added to this data set through database maintenance formatted screens or LOAD or DEMAND commands.

2.2.2 Dataset Data Set

The dataset data set contains different types of members. Data set members define data sets, their attributes and using-jobs known to CA-7. Network members define workstation groupings for related activities. Schedule members contain scheduling criteria for jobs and input/output workstations. User documentation (prose) members are free-form documentation.

Information in the dataset data set is added and maintained through database maintenance. Additionally, LOAD functions add and update data set members.

2.2.3 Index Data Set

The index data set contains creation information for data sets used by CA-7 controlled jobs. Entries are maintained only for the last three creations of any data set. In addition to data set entries, the index data set contains pointers to information stored in the Job and dataset data sets and to event-driven processing relationships.

2.2.4 VRM Data Set

The VRM data set contains information about jobs and the associated resource connections. The resource records maintain information about the jobs use of the resource including the resource name, the resource type (exclusive, shared, address space, corequisite, and resource count resource), the free type (disposition), and step name. Information in the VRM database data set is added and maintained through the virtual resource management facility. The VRM database component is also used as a resource queue by maintaining information about resource use for jobs which have been submitted by CA-7 and for all resource activity.

2.2.5 ARF Data Set

The ARF data set contains three kinds of records that are used for ARF processing: ARFSET records, AAR records, and ARFQ records.

An ARFSET is a named collection of ARF definitions. Each definition provides information for the evaluation of an exception condition and the responses that are to be executed when the exception is detected. Use the AR.3 panel to create and maintain ARFSET records.

An extended copy of the ARFSET record is created or updated any time a job referring to the ARFSET enters the request queue. This is known as an AAR record. The AAR represents the ARFSET that is currently being used to monitor one or more production jobs in the system. ARF dynamically deletes the AAR record when there are no more jobs on the system to be monitored using that ARFSET.

An ARFQ record is created for each occurrence of an exception condition detected by ARF. For example, if five distinct step completion errors are detected for a job, there are five ARFQ records on the ARF data set for that job. ARFQ records are used to schedule the responses associated with the exception conditions in the ARFSET. ARFQ records are dynamically deleted when all responses have been scheduled.

2.2.6 JCL Library(s)

These data sets normally exist prior to installation of CA-7 and are simply defined to CA-7. They contain execution JCL (JOB statement, and so forth) to be submitted to the operating system. PDS, CA-Panvalet, and CA-Librarian organizations are supported. Cataloged procedure libraries (PROCLIBs) are not defined to CA-7 directly but continue to serve the same function as always.

2.2.7 Database Verification

The Database Verification program is a facility which can be used to verify the contents of the database. It includes a logical pointer verification feature that produces error messages for pointers found to be inaccurate. Database Verification is fully defined in the *CA-7 Database Maintenance Guide*.

2.2.8 Backup/Reload

In all production environments, backup and reloading of databases is an extremely important consideration, and backups should be scheduled on a regular basis. For a comprehensive discussion on the CA-7 backup and reload procedures, see 7.1.1, "Backup Procedures" on page 7-2 and 7.1.2, "Reload Procedures" on page 7-11.

2.3 Queues

A group of data sets called the CA-7 queues support automatic scheduling, control, and tracking of work. There are nine such queues. Six are considered status queues. The remaining three queues are called work queues. The following is a brief description of each queue.

2.3.1 Status Queues

Preprocess queue	Tracks input activities for workstations.
Request queue	Contains records of all computer jobs scheduled by CA-7. Jobs may enter the request queue because of automatic scheduling by CA-7 or because of demand scheduling to satisfy a user request.
Ready queue	Holds records of jobs with all requirements satisfied which are waiting to be submitted, or have been submitted, to the operating system for execution.
Active queue	Contains records of jobs which have become active on a computer.
Postprocess queue	Performs a function similar to the preprocess queue for tracking output data handling.
Prior-run queue	Maintains data from the previous execution cycle of each job.

2.3.2 Work Queues

Trailer queue	Contains requirements, load data, step condition code checking data, and JCL for jobs in the request, ready and active queues. JCL retention for jobs in the prior-run queue is a job level user option. It also contains run log data.
Disk queue table (DQT)	Contains control information for all CA-7 online terminal messages.
Scratch queue	Contains response messages for terminals and provides space for CA-7 application work areas.

2.3.3 Work Flow Through the Queues

During processing, work (defined to CA-7) is moved from one CA-7 queue to the next. Appearance of work in a given queue depends on the status of that work at a particular time.

Following is a narrative of the work flow within the CA-7 queue structure. Numbers within parentheses in the narrative refer to circled numbers in Figure 2-2 on page 2-13.

- (1) The preprocess queue is used by CA-7 to track input workstation activity. Work enters the preprocess queue as a result of:

- Schedule scan placing work in the queue because of calendar schedules.
- Demanding of an input network by the user with the DMDNW command.

Within CA-7, a network (input and/or output) is a structure of user-defined workstations (data entry, production control, distribution) where activities related to work processing are performed. A network may or may not be related to a CPU job. For example, a network may be defined, scheduled and invoked to track work flow or control nonproduction or non-CPU data processing activities.

When input networks are used with CPU jobs, these networks define activities to be completed before the associated job executes on a CPU. Networks are scheduled and controlled somewhat like CPU jobs and are referenced by CA-7 in controlling CPU job execution. Prompts are issued to workstations whenever work is late or when defined deadline time (must start times) indicate that work may become late.

- (2) Work (CPU jobs) enters the request queue as a result of:

- Schedule scan placing work in the queue because of calendar schedules.
- A DEMAND command to demand processing of a particular job.
- A LOAD command to cause loading of a job to the CA-7 database.
- A RUN command to run a job independent of predecessor job or data set requirements.
- A job completion, input workstation network completion or data set creation triggering a subsequent job.

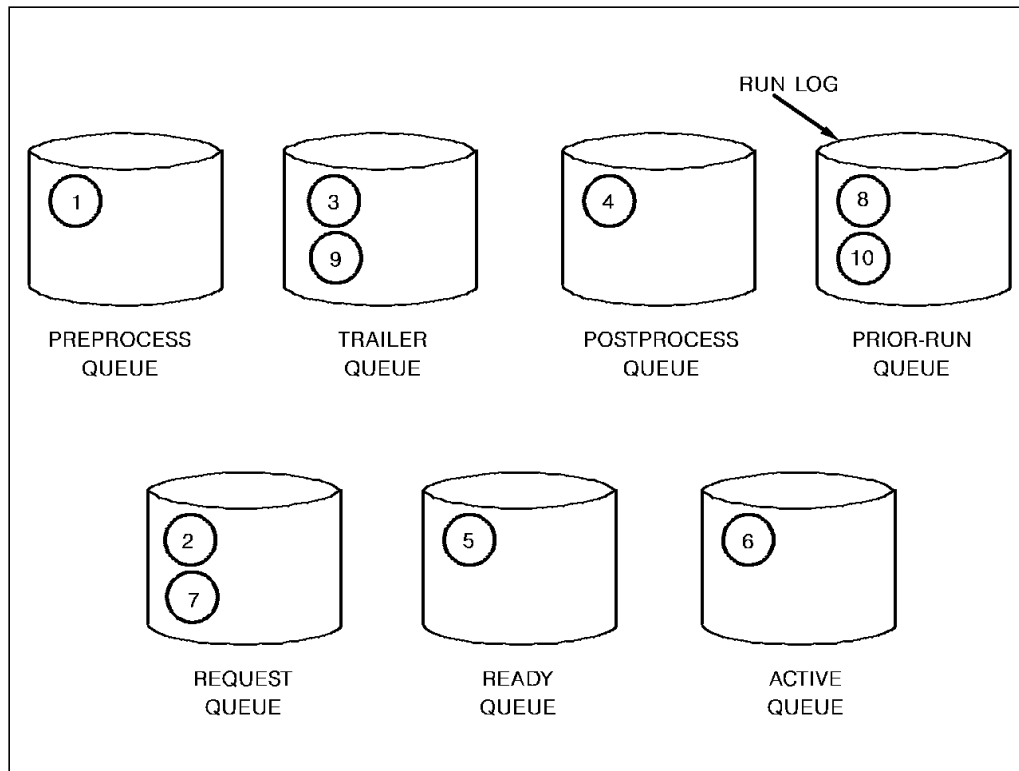


Figure 2-2. Work Flow Through CA-7 Queues

(3) and (4)

At the time a job is placed in the request queue, JCL for the job is added to the trailer queue from the JCL data set. Output networks associated with the job are placed in the postprocess queue. As a job is processed through CA-7, JCL pointers are maintained which point to the data in the trailer queue.

The request queue contains requests by CA-7 (or the user) to run CPU jobs. Each job in the request queue has a master requirements count. This is a count of activities (requirements for the job) which must be completed before the associated job can be sent to a CPU for execution.

These activities may include both internal and external events. Internal events might include creation of an input data set by another job, successful completion of another job, and so on. Internal events are monitored and event completion is posted automatically by CA-7. External events might include JCL overrides, creation of a data set by a job unknown to CA-7, or any other type of manual activity. Completion of external activities is posted manually through a CA-7 command or automatically by batch external programs (U7SVC, BCLP, trailer step). (An external event may be posted by a job submitted by CA-7 which updates the data set, even though the data set is marked external.)

As events are completed and posted, the master requirements count for a job is decremented. When the count goes to zero (when all requirements are satisfied), the job is ready for CPU processing.

Jobs placed in the request queue by a LOAD or RUN command do not have data set or predecessor requirements but may enter the request queue with a HOLD requirement. This requirement is imposed through use of the LOADH (LOAD/HOLD) command or the RUNH (RUN/HOLD) command. Jobs entering the request queue with a HOLD requirement must be manually released from the queue by posting the requirement.

- (5) When all requirements for a job are satisfied, the job is moved to the ready queue.
- a If the workload balancing feature is not turned on, the JCL necessary to execute the job is written to the internal reader or submit data set (for the CPU on which the job is to execute).
 - b. If the workload balancing feature is active, the JCL is not written to the internal reader or data set until the job satisfies the job mix optimization criteria specified by the user.
 - c. If the job has any resource connections (through Virtual Resource Management), a check is made to determine if the resources are available and meet the execution requirements for the job. If the resource requirements are satisfied, the JCL is written to the internal reader or submit data set. If there are resources which are unavailable to the job, the job stays in the ready queue with a status of W-RSRC - waiting on resources.

(6) and (7)

When a job becomes active on a CPU, the SMF type 20 or 30 record (job initialization) for the job is received by CA-7 and the job is moved to the active queue. The job remains in the active queue until an SMF type 5 or 30 record (job completion) is received by CA-7. On receipt of the job completion record, the job is moved back to the request queue for job completion processing.

If CA-7 determines, through condition or abend codes, that a job did not successfully complete, the job is moved from the active queue to the request queue where it remains until it is restarted or canceled. At this time, a restart prompt is issued for the job. A list of all jobs requiring restart is available through the LIST command.

During job completion processing, the following will happen:

- Data sets created by a successfully completed job are updated in the index data set and used to post requirements for those data sets used by other jobs in the request queue.
- During completion processing, job triggers and data set triggers are examined to determine if additional jobs should be brought into the request queue.
- Requirement records in the database for NEXT-RUN=ONLY are discarded.
- Requirement records in the database for NEXT-RUN=SKIP are reset to YES.

(8) and (9)

Once completion processing is finished, the status queue record is placed in the prior-run queue. This serves as a last runtime record for the job. Optionally, JCL used to execute the job may be retained in the trailer queue. Retaining JCL in the trailer queue has the advantage of having JCL available for review and reuse should the need arise. However, the practice of retaining a large volume of JCL will impact the size requirements of the trailer queue.

(10)

For every job completion, job cancellation, job restart, completion of a preprocess or postprocess network, or cancellation of a preprocess or postprocess network, an entry is written to the run log which is referenced by a prior-run queue record. This log is accumulated continuously with a user specified number of days being retained (the default is 5 days).

2.4 Other Data Sets

The following is a brief description of data sets not considered part of the database or queue structures previously described.

Checkpoint data set Used by CA-7 to record checkpoints for restart purposes.

Communications data set

A collection and transfer point for CA-7 and ICOM information. The data set provides an area for ICOM to store SMF and trailer data. Also used for batch terminal communications with CA-7.

Submit data set(s) Optional. Contains image JCL. JCL is read from this data set by ICOM and passed to the host system. There may be multiple submit data sets. In a shared spool MVS environment, submit data sets can be replaced with an internal reader.

Log data set(s) Stores SMF data collected by ICOM, messages that were written to the browse data set, records of certain events within CA-7 and commands to CA-7. One or two log data sets may be used. When two are used, one is considered primary and one secondary. When one becomes full, CA-7 automatically switches recording to the other and submits a job to dump the log data set which has become full.

Log history/archive data set

Contains CA-7 Log records as they were output by the SASSHIS5 and SASSHIS6 history management jobs. It is recommended that this data set reside only on tape due to the volume of records accumulated.

Queue dump data set Used during a DMPQ/MOVQ operation to temporarily hold the queue data being moved.

VRM dump data set Used during a DMPQ/MOVQ operation. This file contains the VRM active resource records which correspond to jobs in the CA-7 queues.

Batch terminal data sets

Defined as input/output pairs of DASD data sets and used for batch processing of CA-7 commands.

Statistics data set Contains statistics on day-to-day system operation for up to a two-year period.

Workload planning data sets

Optional. Used to hold WLP input data created from the database with the FWLP command.

Browse data set	An optional sequential, wraparound print line image data set that can be used as an online browsing alternative to a master station printer.
Help library	Contains reference information about CA-7 applications, with general and detailed descriptions as well as sample commands.
Override library	Used for temporary versions of a job's JCL. Must also be defined with a JCL statement in the initialization file.
Options data set	Contains the runtime options for CA-JCLCheck. This data set is optional and is used only during initialization if the CA-7 CA-JCLCheck interface is available.
CA-7 JCLLIB	Created during CA-7 system generation. This PDS contains installation and maintenance jobs and the CA-7 initialization file which defines the environment and configuration for running CA-7. It also contains the DBPARMS data set which defines the VSAM data sets for the database.
CA-Driver PROCLIB	Contains JCL procedures which are used by CA-Driver for JCL tailoring.
Model CA-Netman transaction file	Contains schema that are used to build CA-Netman API transactions issued by the realtime CA-Netman interface. This data set is required if the NETMAN initialization file statement is present.
CA-Netman API deferred request data set	This data set contains those CA-Netman API transactions issued by the realtime CA-Netman interface when the CA-Netman CCI receiver was inactive.
CA-7 calendar PDS	This optional data set contains copies of CA-7 base calendars. It is required only if you wish to use online facilities to create and maintain CA-7 base calendars.
CA-7 Cross-Platform Scheduling Profile	Contains environment variables for CA-7 Cross-Platform Scheduling.
CA-7 Cross-Platform Checkpoint File	Contains checkpoint information used by the CA-7 Cross-Platform Tracking System (XTRK) to coordinate tracking functions in non-OS/390 platforms.

2.5 Terminal Communications

Terminal communication with CA-7 is provided through a terminal network. The network provides both online and batch access to CA-7 facilities.

2.5.1 Logical Terminals

A logical terminal is a symbolic name, known as station name, assigned to a real (physical) terminal defined to the CA-7 system. A single device may represent multiple logical terminals to the CA-7 system. For example, one physical terminal might service several functions. Each functional use could be assigned its own logical name.

2.5.2 Online Access

Online access (versus batch access) is provided by 3270-type terminals (and optionally associated printers). These terminals are used for input of CA-7 commands, and for output of CA-7 messages and prompts. OS consoles are also considered to be online terminals.

The CA-7 master terminal (CONS=MASTR in the CA-7 initialization file) must be an input and output device and may be the OS console or any online terminal. In addition to serving as the master console, the terminal may be used to input commands and data like any other terminal in the network. Normally, very few CA-7 messages should be received by the master console during operation.

The master station may be a browse data set, or a 3270-compatible terminal or printer. It may be defined as the same physical device as the master console, although this is not recommended. Unlike the master console, the master station receives the majority of messages issued by CA-7 during job processing. If the master console and the master station are to be defined as the same physical device, an OS console cannot be used.

Note: The master terminal (CONS=MASTR in the CA-7 initialization file) should not be confused with the master station (STANIDS=MASTER).

2.5.3 Batch Access

Batch access to CA-7 is provided by several programs. Each of these is briefly described below. Additional information is available in the "External Communicators" chapter of the *CA-7 Interfaces Guide*.

2.5.3.1 Batch Terminal Interface

Batch terminals provide the ability to process commands in batch mode as if from a real terminal. Up to eight batch terminals may be assigned to CA-7. Each batch terminal consists of a pair of disk data sets, one input, one output. These data sets are defined to CA-7 so they appear to be a real terminal. Use of a batch terminal is accomplished by using the CA-7 batch terminal interface program.

2.5.3.2 U7SVC Facility

U7SVC is a program which combines certain functions of both the trailer step and the Batch Card Load program. U7SVC may be executed as a job step in batch or be called from a user program. U7SVC may be executed on any computer where ICOM is active. The user may use U7SVC to post to CA-7 the creation of a data set or to issue CA-7 batch commands.

2.5.3.3 Trailer Step

Trailer steps are special job steps which execute a utility type program. They may be added to any job (including jobs not defined to CA-7) to cause the processing of CA-7 queue maintenance posting commands. A trailer step may be executed on any computer where ICOM is active. Although it does cause activity within CA-7, it does not affect the operational function of the job which contains the trailer step. The trailer step may be used to perform any of the queue maintenance posting functions as defined in the *CA-7 Commands Guide* as well as some other CA-7 functions.

2.5.3.4 Batch Card Load Program (BCLP)

The Batch Card Load program loads card-image data to required data sets for CA-7 jobs. Through BCLP, single or multiple data sets may be created, replaced or modified. When BCLP is executed, CA-7 is notified of successful data set manipulation through the communications data set. In this way, the database can be updated to reflect the creation date and time. In addition, the request queue is scanned for jobs which have that particular data set as an input requirement. BCLP may be executed on any computer where ICOM is active.

2.5.3.5 Internal Terminals

Internal terminals are used by CA-7 system functions such as Cross-Platform Scheduling and the Automated Recovery Facility (ARF). These terminals allow work to be scheduled on behalf of threads not directly associated with online terminal users or tasks using the external communicators such as trailer steps. For example, ARF uses an internal terminal to issue CA-7 commands in response to exceptions detected.

2.5.3.6 CCI Terminals

CCI terminals are used to schedule work in CA-7 from an external task such as a batch job or a REXX exec running in a TSO session. The format of CCI terminal input and output is the same as that used by the Batch Terminal Interface.

2.5.4 BSAM Terminal (Browse Data Set)

A BSAM terminal is a sequential DASD file which can be used as an alternative to a terminal printer. This file is treated as a wraparound, 80-character print image data set. The primary use of a BSAM terminal is for master station messages.

All master station messages are written to the log data set whether or not a browse data set is being used. If a browse data set is defined with the appropriate GROUP, LINE, and TERM statements in the initialization file, messages are also written to this sequential, wraparound, 80-character, print line image data set. The data set is allocated by the user to contain as much message volume as desired for online browsing. Once the allocated area has been filled with messages, writing of messages to the data set begins again at the front of the allocated area.

At startup time, the browse data set is opened as a sequential output data set. No check-pointing is done for the data set. Therefore, message traffic is written beginning at the front of the area allocated. If a restart of CA-7 is being done and there are previous messages which the user wants to keep from being overlaid, the user must preserve the data, prior to restarting CA-7, using whatever technique satisfies their particular requirements. A simple copy of the data set to another DASD area, or to a magnetic tape, could quickly save the data.

If browsing is to be done through TSO, CA-Roscoe, or any facility other than CA-7, the DISP parameter in the DD statement must be coded as SHR for shared access.

There is a special onetime consideration whenever the browse data set is being used for the first time. Since the data set is used in a wraparound manner, formatting occurs automatically while messages are being written during execution of CA-7. However, until the area has been completely used the first time, reading beyond end-of-file would occur during any attempt to browse the data set unless the area had been preformatted by the user.

Chapter 3. Installation Requirements

This chapter presents several considerations related to the installation of CA-7. Attention is focused on the operating system and job entry systems that are supported, data set requirements, security, calendar definition and structure, workload balancing, and interfaces with other products.

3.1 Operating Systems and Job Entry Systems Supported

CA-7 operates under OS/390, MVS/XA, and MVS/ESA operating systems.

CA-7 operates with JES2 or JES3 job entry subsystems.

3.1.1 Features Required

- VTAM support for 3270 terminal access.
- SMF type 14 (optional), 15, 26, and 30 records (or optionally type 4, 5, and 20 instead of 30s).
- One Type IV SVC.
- CA-7 should execute in standard problem program protect key (usually protect key 8).
- CA-7 installation and maintenance require SMP/E.
- In your JCL, columns 70 and 71 of the JOB statement must be blank for any job to be submitted by CA-7. If external security with USERID insertion is used, column 69 must be a blank or a comma.

If CA-7 NCF is used, columns 69, 70, and 71 of the JOB statement must be blank for **ANY** job to be submitted by CA-7. If external security with USERID insertion is used, column 68 must be a blank or a comma.

- A JCL INCLUDE statement cannot be used for the JOB statement or the first EXEC statement.

3.1.2 Memory Requirements

On one host CPU, the CA-7 Central Control System requires a minimum of 4M of virtual storage. However, as the use of CA-7 increases, so do the virtual storage requirements. See 6.1.1, “REGION Size” on page 6-2 for further discussion.

On each CPU where CA-7 controlled jobs execute, ICOM requires approximately 64K of virtual storage. An ICOM is required for each processor generating its own SMF records.

Note: Because CA-7 issues a SYSEVENT DONTSWAP, some of the LSQA pages can be flagged as long term page fixed. If the machine is to be partitioned, then CA-7 would have to be recycled across the split. To prevent the necessity of recycling CA-7 on ESA or OS/390 systems, the SYS1.PARMLIB SCHEDxx member can be updated. Simply add a PPT entry for CA-7 to indicate SPREF and LPREF. For XA systems, the PPT would need to be updated to show CA-7 as nonswappable. If partitioning is not going to be done, then no problem should exist and no updates are required.

3.2 Data Set Requirements

This topic describes each CA-7 data set as to its contents, organization, space requirements, and DASD restrictions. Table 3-1, summarizes the data set requirements.

Table 3-1 (Page 1 of 2). Summary of Data Set Requirements				
Data Set	Organization	Online	Est. Space on 3380-I	DASD Restrictions
Job data set	VSAM	yes	variable (note 2)	none
Dataset data set	VSAM	yes	variable (note 2)	none
Index data set	VSAM	yes	variable (note 2)	none
VRM data set	VSAM	yes	variable (note 2)	none
ARF data set	VSAM	yes	variable (note 2)	none
JCL data sets	PO/LIB/PAN	yes	variable (note 2)	none
Preprocess queue	PSU	yes	5 tracks	(note 3)
Request queue	PSU	yes	10 tracks	(note 3)
Ready queue	PSU	yes	5 tracks	(note 3)
Active queue	PSU	yes	5 tracks	(note 3)
Postprocess queue	PSU	yes	5 tracks	(note 3)
Prior-run queue	PSU	yes	variable (note 2)	(note 3)
Trailer queue	PSU	yes	variable (note 2)	(note 3)
Disk queue table queue	PSU	yes	variable (note 2)	(note 3)
Scratch queue	PSU	yes	variable (note 2)	(note 3)
Checkpoint data set	PSU	yes	1 cyl	Same device as queues
Communications data set (note 4)	PS	yes	minimum 50 tracks	(note 2)
Submit data set(s)	PS	yes	1 cyl	none
Log data set(s)	PS	yes	At least 10 cyls	none
Log history/archives	PS	no	variable (note 2)	none
Queue dump data set	PS	yes	variable (note 2)	none
VRM dump data set	PS	yes	variable (note 2)	none

3.2 Data Set Requirements

Table 3-1 (Page 2 of 2). Summary of Data Set Requirements				
Data Set	Organization	Online	Est. Space on 3380-I	DASD Restrictions
Batch terminal data sets	PS	yes	input: 5 tracks output: 10 cyls	none
Statistics data set	PS	yes	750 records of 1024 bytes each	none
Workload planning data sets	PS	yes	variable (note 2)	none
Browse data set	PS	yes	variable (note 2)	none
HELP library (note 1)	PO	yes	variable (note 2)	none
Override library	PO/LIB/PAN	yes	variable (note 2)	none
CA-7 JCLLIB	PO	yes	1 cyl	none
CA-JCLCheck options data set	PS/PO	yes	variable (note 5)	none
CA-Driver procedure library	PO	yes	variable (note 6)	none
Model CA-Netman transaction file	PS	yes	1 track (note 7)	none
CA-Netman API deferred request	PS	yes	variable (note 7)	none
CA-7 calendar PDS	PO	yes	variable (note 2)	none
CA-7 Cross-Platform Scheduling Profile	PO	no	variable (note 2)	none
CA-7 Cross-Platform Checkpoint file	PS	no	1 track (note 2)	none

* All data sets MUST be CONTIG except for the JCL, override, and CA-7 JCLLIB libraries.

Note 1: Space requirements are specified in the *CA-7 Getting Started* guide.

Note 2: See the individual discussions later in this chapter.

Note 3: 3330 (Mod I or II), 3350, 3375, 3380, 3390, 9345. All queues and checkpoint data set must be on the same device type.

Note 4: Shared DASD if multi-CPU.

Note 5: Used only if desired for the CA-7 CA-JCLCheck interface.

Note 6: Used only if CA-Driver component is desired.

Note 7: Required with the realtime CA-Netman interface.

3.2.1.1 Target Libraries for CA-7

The following table shows the CA-7 SMP target libraries. The *CA-7 Getting Started* guide discusses space requirements.

Name	Description
CAI.CA7.CAICLIB	CLIST library
CAI.CA7.CAIISPP	ISPF panel library
CAI.CA7.CAIISPT	ISPF table library
CAI.CA7.CAILIB	Load library
CAI.CA7.CAIMAC	Macro library
CAI.CA7.CAISRC	Source library
CAI.CA7.PPOPTION	Options library

3.2.2 Database Space Requirements

Space requirements are determined by the IDCAMS parameters specified, in the UCC7DBASE statements in the DBPARMS member of the CA-7 JCLLIB, when the database is loaded. The record lengths used are based on the RECORDSIZE values specified in the IDCAMSDEFINE parameters at reload time.

The user can select any higher blocking size up to the maximum of 32760 bytes. The maximum record size must not be smaller than 3122. The recommended average and maximums are:

Average - 60 bytes

Maximum - 4080 bytes

A CI size of 4096 is also recommended. These can be varied by the individual user to best meet the needs for DASD space, memory requirements, and performance levels based on the individual situation and experience.

3.2.3 Database Files

The CA-7 database is comprised of the job data set, the dataset data set, the index data set, the VRM data set, the ARF data set, and the JCL file(s). JCL file(s) usually exist prior to installation of the system. JCL file(s) contain actual execution Job Control Language statements. The remaining four files are unique to CA-7:

- the job data set contains CPU job structure and relationship data.
- the dataset data set contains data set characteristics and usage data as well as workstation and network definitions, all schedules and documentation.
- the index data set contains data set version information, system cross-reference data and triggering relationships.
- the Virtual Resource Management (VRM) data set contains job to resource connection records for jobs using resources under CA-7.
- the Automated Recovery Facility (ARF) data set contains information on detection and handling of exception conditions for production jobs.

The database may reside on any direct access device supported by the operating system.

3.2.3.1 Job Data Set

Contents: Data that reflects the structure, dependency relationships, execution characteristics and data set use.

Organization: Built and maintained in alphanumeric sequence by job name. This file is a key-sequenced VSAM data set with characteristics described under 7.1.4.3, “IDCAMSDEFINE Parameters” on page 7-19.

Default Space Requirements: The installation process sets the default space allocation for the job data set at CYL(10 1). This size should be closely monitored during the initial implementation phase to see if it needs to be increased. Also, as the workload grows, the space allocation may need to be increased.

DASD Restrictions: None.

3.2.3.2 Dataset Data Set

Contents: The dataset data set contains data set records, documentation records, workstation network records, and schedule records. It provides information about every data set referenced by CA-7 controlled jobs. Database information and using-job pointers are built automatically. User entered data includes documentation (prose) and schedules used to control prompting for input/output and execution deadlines.

Organization: The dataset data set is in sequence by DS number which is dynamically assigned within data set type prefix. Access to the data set is possible by data set name through cross-reference pointers from the index data set or directly by data set number. This file is a key-sequenced VSAM data set with characteristics described under 7.1.4.3, “IDCAMSDEFINE Parameters” on page 7-19.

Default Space Requirements: The installation process sets the default space allocation for the dataset data set at CYL(15 1). This size should be closely monitored during the initial implementation phase to see if it needs to be increased. Also, as the workload grows, the space allocation may need to be increased.

DASD Restrictions: None.

3.2.3.3 Index Data Set

Contents: The index data set (IDS) contains basically the same volume information as the OS catalog but reflects only those data sets defined to CA-7. In addition to volume entries, IDS contains entries for all documentation members and all networks. All entries in the IDS contain cross-reference pointers which allow a reference back to the job and dataset data sets.

Organization: This file is a key-sequenced VSAM data set with characteristics described under 7.1.4.3, “IDCAMSDEFINE Parameters” on page 7-19.

Default Space Requirements: The installation process sets the default space allocation for the index data set at CYL(3 1). This size should be closely monitored during the initial implementation phase to see if it needs to be increased. Also, as the workload grows, the space allocation may need to be increased.

DASD Restrictions: None.

3.2.3.4 VRM Data Set

Contents: The Virtual Resource Management (VRM) data set contains job to resource connection records for jobs using resources under CA-7. The VRM data set is also used as a queue for job resource management during job submission and job/step completion.

Organization: This file is a key-sequenced VSAM data set with characteristics described under 7.3.3.1, “VRM Data Set IDCAMS Define Parameters” on page 7-32.

Default Space Requirements: The installation process sets the default space allocation for the VRM data set at CYL(5) for the DATA component and TRK(5) for the INDEX component.

DASD Restrictions: None.

3.2.3.5 ARF Data Set

Contents: The Automated Recovery Facility (ARF) data set contains information on the detection and handling of exception conditions for production jobs.

Organization: This file is a key-sequenced VSAM data set with characteristics described in 7.2.3.1, “ARF Data Set IDCAMS Define Parameters” on page 7-27.

Default Space Requirements: The installation process sets the default space allocation for the ARF data set at CYL(5) for the DATA component and TRK(5) for the INDEX component.

DASD Restrictions: None.

3.2.3.6 JCL Data Sets

These data sets contain the execution JCL for the jobs defined in the CA-7 database. When jobs are submitted by CA-7, the JCL submitted is taken from these data sets. PDS, CA-Panvalet, and CA-Librarian organizations, in any combination, are supported. Refer also to the discussion of the override library.

For a JCL library whose organization is PDS, LRECL must be 80.

CA-7 has no other unique requirements of these data sets. Space allocations should be whatever is required to satisfy the needs of the user.

In a CA-7 environment which uses UID security, it may be desirable to have a separate JCL library for each UID. See the UID macro in the *CA-7 Security Guide* and the UID field on the DB.1 screen for related discussions of this security facility.

See the initialization file JCL statement for the required definitions of JCL data sets. (See 5.2.24, "JCL Statement" on page 5-57.)

The "JCL Management" chapter of the *CA-7 Database Maintenance Guide* and the General Inquiry Facility topic of the *CA-7 Commands Guide* also contain discussions of other CA-7 functions related to accessing JCL.

3.2.4 Queue Files

The CA-7 dynamic scheduling and tracking of work is supported by the queue data sets. There are six queues which contain status information and three work queues.

3.2.4.1 Status Queues

The status queues are as follows:

- Preprocess queue
- Request queue
- Ready queue
- Active queue
- Postprocess queue
- Prior-run queue

The preprocess queue is used for tracking input activities from workstations which may be used to satisfy input execution requirements of jobs.

The request queue contains all CPU jobs scheduled by CA-7. Jobs may enter the request queue because of automatic scheduling by CA-7 or because of demand scheduling to satisfy a user request.

The ready queue holds jobs with satisfied requirements which are waiting to be, or have been, submitted to an operating system for execution.

The active queue contains jobs which are active on a CPU.

The postprocess queue performs a function similar to the preprocess queue for tracking output data.

The prior-run queue maintains data from the last successful execution cycle of each job. This data can be used to facilitate restart/rerun activity.

3.2.4.2 Work Queues

The work queues are as follows:

- Trailer queue
- Disk queue table (DQT)
- Scratch queue

The trailer queue contains requirements, output data, step condition code checking data, load data, and JCL for jobs in the request, ready, and active queues.

The disk queue table (DQT) contains message controls for all CA-7 online and batch terminals.

The scratch queue contains messages for terminals and provides space for CA-7 scratch work files.

3.2.4.3 Organization

All CA-7 queues are allocated as physical sequential data sets with a block size of 1024 bytes. Access to these files, however, is through the CA-7 queue access method, which uses a technique of dynamic track allocation. These must be allocated as permanent and unmovable.

Note: As a one time (temporary) run, the queues can be allocated as temporary instead of permanent, but it requires a FORM start and no queue data is saved across a stop and start of CA-7.

DCB=(DSORG=PSU,RECFM=F,LRECL=1024)

3.2.4.4 Space Requirements

Specific space requirements are based on an IBM 3380 unless otherwise specified. Space allocations for each queue must be contiguous and unmovable. The space values indicated below are only recommendations for a starting point. The values may need to be changed later. The command /DISPLAY,Q=ALL or the CA-7 Queue Allocation Usage report can be used to monitor the space use. None of the queues should be allocated with less than 3 tracks.

PREQ	If input networks are not used, allocate 3 tracks. Otherwise, allocate 20 tracks.
REQQ	Allocate 20 tracks.
RDYQ	Allocate 10 tracks.
ACTQ	Allocate 5 tracks.
POSTQ	If output networks are not used, allocate 3 tracks. Otherwise, allocate 20 tracks.
PRNQ	The prior-run queue contains one entry for each CPU job run under CA-7 control. There is space for five job entries in each 1024-byte block. This data set should have minimum space allocation of 1 entry for each job defined in the database.
TRLQ	At least 2 tracks are required during the queue life of each job in the request, ready, and active queues. (Requirements use at least 1 track per job, and the execution JCL also uses 1 track). At least 1 track per job is retained for each job for which JCL is saved through the prior-run queue.
DQTQ	Allocate 2 tracks per terminal for the first 50 terminals defined in the initialization file. If more than 50 terminals are defined, allocate one track for each terminal over 50.
SCRQ	Each queued message uses at least 1 track in the scratch queue. Allocate a minimum of 300 tracks. High terminal activity may require that this be increased.

The DQTQ, SCRQ, and TRLQ are used by track. Therefore, if 400 tracks are allocated on a 3350-type device, then 400 tracks would also be required to move the queue to a 3380-type device.

3.2.4.5 DASD Restrictions

CA-7 queues and the checkpoint data set must reside on the same device type.

3.2.5 Other Online File Requirements

The following data sets discussions complete the data set list for CA-7 (not all are required). Except for the source and macro libraries, all of these data sets must be available to the CA-7 online system.

3.2.5.1 Load Library

Contents: CA-7 executable programs.

Organization: Standard SMP target load library.

DCB=(RECFM=U)

Space Requirements: See the *CA-7 Getting Started* guide for estimates.

3.2.5.2 Checkpoint Data Set

Contents: Used to record checkpoints for restart purposes.

Organization: Direct data set accessed through EXCP.

DCB=(DSORG=PSU)

Space Requirements: One cylinder (CONTIG), unmovable. A minimum of 10 tracks suffices, but allocation must begin on a cylinder boundary. The checkpoint data set is unmovable and must reside on the same device type as the queues. If the checkpoint data set is reallocated, one of the COLD types of starting CA-7 must be done.

DASD Restrictions: Must be on the same device type as the queues.

3.2.5.3 Communications Data Set

Contents: Used as a collection and transfer point for CA-7 data and control information. In single and multiple CPU configurations, the data set provides an area for the Independent Communications Manager (ICOM) to store SMF and trailer data.

Organization: Similar to that of the queue data sets. The access method is EXCP.

```
DCB=(RECFM=F,BLKSIZE=1024,LRECL=1024)
```

Space Requirements: Space required depends on the number of jobs controlled by CA-7. The communications data set requires a minimum of 50 contiguous tracks.

DASD Restrictions: Must reside on a shared DASD device in multi-CPU environments.

Formatting the Communications Data Set: Before ICOM can use the communications data set, the data set must be formatted by the SASSCDSI program. This formatting process is required if the size or structure of the communications data set is to be changed, the data set is moved, or if a new submit data set is being defined to extend CA-7 control over another CPU.

If formatting of the communications data set has not been done prior to execution of ICOM, the following JCL should be used to accomplish the required formatting. (See the N700 install job.) For NCF concerns, see DASD requirements in Chapter 4 of the *CA-7 Interfaces Guide*.

```
//stepname EXEC PGM=SASSCDSI,REGION=256K
//STEPLIB DD DSN=user-defined-CA-7-loadlib,DISP=SHR
//SYSPRINT DD SYSOUT=A
//UCC7CMDS DD DSN=communications-data-set,DISP=OLD
//SYSIN DD *
*CDSINIT*
*ICOMCPU* xxxxyyyzzzzaaaa
*CPU* xxxxxxxx
/*
```

SASSCDSI SYSIN Data Set: There are three types of SYSIN control statements required for initialization of the communications data set. These statements are as follows:

```
*CDSINIT*
```

This must be the first statement in the input stream and must begin in column 1.

```
1      11
*ICOMCPU* xxxxyyyzzzzaaaa
```

This must be the second statement in the input stream and must begin in column 1.

xxxx, yyyy, zzzz and aaaa are the system IDs defined to SMF through the appropriate SYS1.PARMLIB member. Up to four CPUs may be specified. They must begin in column 11. Each system ID must be 4 characters long. Multiple IDs may be specified in any sequence. Systems IDs are used for documentation purposes only. If ICOM is active on more than four systems, only four can be specified. This does not affect the processing of ICOM or of CA-7.

```
1      11
*CPU*   xxxxxxxx
```

Defines the submit data set for each ICOM (or CPU). One *CPU* statement is required for each CPU to be controlled by CA-7. The *CPU* entry must begin in column 1. The xxxxxxxx value must begin in column 11 and specify the ddname of a submit data set as defined in the CA-7 JCL and the ICOM JCL. If the internal reader option is taken (see below), at least one *CPU* statement is still required.

3.2.5.4 Submit Data Set(s)/Internal Reader(s)

Contents: In a nonshared spool environment, there must be a submit data set for each job entry subsystem and the data set must reside on a device accessible by CA-7 and the ICOM submitting JCL from that data set. In a shared spool or single CPU environment, jobs can be submitted by CA-7 directly to an internal reader instead of submitting it to a submit data set. See the initialization file CPU statement DDNAME parameter for more on this option. (See 5.2.22, “CPU Statement” on page 5-54.) If the internal reader option is not taken, this data set contains card-image JCL to be submitted by ICOM to the host (JES) system. The data functions as a reader queue or transfer queue between CA-7 and the operating system's internal reader.

Note: The maximum number of submit data sets that can be supported by CA-7 is 6.

Organization: A physical sequential data set containing 80-byte records. BLKSIZE must also be 80 bytes.

DCB=(RECFM=F,LRECL=80,BLKSIZE=80)

Space Requirements: One cylinder contiguous space or enough space to hold the JCL for the largest job controlled by CA-7. There are no space requirements for an internal reader.

DASD Restrictions: None.

3.2.5.5 Log Data Set(s)

Contents: Physical sequential data set(s), used to store SMF data collected by ICOM. Contains records of certain events within CA-7 and messages to CA-7. Two DASD data sets are allocated and CA-7 alternates their use automatically much like SMF data sets. See the JOB parameter in the DBASE statement in the initialization file. (See 5.2.17, “DBASE Statement” on page 5-43.)

Organization: Physical sequential.

`DCB=(RECFM=VB,BLKSIZE=2104,LRECL=2100)`

Note: BLKSIZE may be increased; however, CA-7 only writes a maximum block of 2104 bytes.

Space Requirements: Space required for log data set(s) depends on the number of jobs run under control of CA-7, the amount of data to be recorded and the frequency with which backups are to be done. There may be a maximum of two log data sets. Suggested space allocation is at least 10 cylinders per log data set. The space must be allocated as contiguous.

DASD Restrictions: Both log data sets must reside on the same DASD volume. The log data set may reside on tape (in unblocked format) if available for permanent allocation to CA-7; however, this is not recommended.

3.2.5.6 Queue Dump Data Set

Contents: Used to temporarily hold the contents of the queue data sets during a DMPQ/MOVQ operation. The queues involved here are:

- Active
- Preprocessing
- Prior-Run
- Ready
- Request
- Postprocessing
- Trailer

The space requirement can be calculated by summing the space allocations for the preceding queue data sets. This sum is usually an extreme overallocation but works.

This data set is referenced by //UCC7QDMP DD in the CA-7 execution JCL.

Organization: The data set is physical sequential with variable-length blocked records.

DCB=(RECFM=VB,LRECL=256,BLKSIZE=256 * n + 4)
--

Note: n is a user-defined value.

Space Requirements: Variable. Must be contiguous and large enough to hold the data from all of the queues previously itemized under Contents.

DASD Restrictions: Must reside on disk.

3.2.5.7 VRM Dump Data Set

Contents: Used to temporarily hold the VRM active resource records which correspond to jobs in the CA-7 queues. Accessed during the DMPQ/MOVQ operation.

This data set is referenced by //CA7VDMP DD in the CA-7 execution JCL.

Organization: The data set is physical sequential with variable-length blocked records.

DCB=(RECFM=VB,LRECL=32700,BLKSIZE=32704)
--

Space Requirements: Variable. Must be contiguous and large enough to hold the VRM active resource information for all jobs in the request, ready, and active queues.

DASD Restrictions: Must reside on disk.

3.2.5.8 Batch Terminal Data Set(s)

Contents: Input and output data set pairs which appear as terminals to CA-7. The input data set contains commands for CA-7 applications. The output data set contains responses from those applications.

Organization: The batch terminal data sets are physical sequential. The input data set is an 80-byte unblocked or blocked file. The output data set is variable blocked with a maximum record length of 137.

Input data set: DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

The block size on the input data set must match BUFSIZE in the LINE statement in the initialization file.

Output data set: DCB=(RECFM=VB,LRECL=137,BLKSIZE=137 * n + 4)

Note: n is a user-defined value.

Space Requirements: The input data set requires 5 tracks minimum. The output data set requires 10 cylinders minimum and should be allocated in a single extent.

These allocations are recommendations only. Actual requirements may be more or less depending on intended use of the batch terminal. Up to eight batch terminal pairs of data sets may be defined.

DASD Restrictions: None.

3.2.5.9 Statistics Data Set

Contents: Contains records with statistics related to system operation covering a two-year period. Used to produce automated performance analysis graphs.

Organization: A physical sequential data set containing 1024-byte records. BLKSIZE must also be 1024 bytes.

```
DCB=(RECFM=F,LRECL=1024,BLKSIZE=1024)
```

Space Requirements: Contiguous space for 750 records of 1024 bytes each is required for this data set.

DASD Restrictions: None.

3.2.5.10 Workload Planning Data Set(s)

Contents: Contains records with job and resource data created by the FWLP command. These data sets are selected individually with the DDNAME keyword of the FWLP command. Any number of these may be defined to CA-7 with their own DD statement in CA-7 execution JCL. Those defined are then available for FWLP commands.

Organization: Physical sequential data set(s) containing card-image, fixed-format records. This file should be allocated in a single extent.

```
DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
```

Note: n is a user-defined value.

Space Requirements: The size of the data set(s) varies according to the amount of data included in the FWLP forecast time span. Space must be allocated before CA-7 is started. Depending on the efficiency of the block size chosen, allocate at least 1 cylinder for each data set.

DASD Restrictions: None.

3.2.5.11 Browse Data Set

Contents: Contains records of print-image messages produced for the master station. Each record contains the equivalent of one print line, just as it would appear on a printer device.

Organization: Physical sequential with 80-character, fixed-length records.

DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

Note: n is a user-defined value.

A large block size can cause messages to only be buffered and not written to the browse data set in a timely manner. Too small a block size can cause performance problems (and bad use of DASD space) due to the number of messages going to browse. A block size of at least 1600 is recommended.

Space Requirements: Size of the data set(s) varies according to the frequency of messages and the period of time desired to be available for online browsing. Whenever end-of-extent is reached during execution, space allocated is automatically reused from the beginning.

DASD Restrictions: None.

Note: This file should be allocated and defined to CA-7.

3.2.5.12 HELP Library

Contents: Application descriptions, command formats, and sample commands for online reference.

Organization: Standard partitioned data set.

DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

Space Requirements: The size may increase slightly for new versions of CA-7. Allocation of 60 tracks and 10 directory blocks should be sufficient.

DASD Restrictions: None.

3.2.5.13 Override Library

Contents: Contains execution JCL which the user places here on a temporary or as needed basis. The JCL index value reserved for the override library is 254. See the USE-OVRD-LIB field on the DB.1 screen for further information.

Organization: Partitioned Data Set (PDS), CA-Panvalet, or CA-Librarian. CA-Panvalet and CA-Librarian data set members are not deleted after the job runs.

Space Requirements: Variable depending on amount and frequency of use as well as the size of run streams. Recommended minimum allocation is 2 cylinders.

DASD Restrictions: None.

Note: This file should be allocated and defined to CA-7.

3.2.5.14 CA-JCLCheck Options Data Set

Contents: Contains runtime options for CA-JCLCheck.

Organization: Sequential card-image file or PDS member.

Space Requirements: Minimal, but varying by number of statements used to define CA-JCLCheck runtime parameters.

DASD Restrictions: None.

3.2.5.15 CA-Driver Procedure Library

Contents: Contains CA-Driver procedures which are added manually.

Organization: Partitioned Data Set (PDS).

DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

Space Requirements: Dependent on the number and size of the procedures used.

DASD Restrictions: None.

3.2.5.16 CA-7 JCLLIB Data Set

Contents: This data set is created during CA-7 system generation to contain CA-7 installation and maintenance jobs, the CA-7 initialization file which defines the environment and configurations for running CA-7, and the DBPARMS deck which defines the database.

Organization: Partitioned Data Set (PDS).

DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

Space Requirements: One cylinder.

DASD Restrictions: None.

3.2.5.17 Model CA-Netman Transaction File

Contents: Contains model CA-Netman API transactions. Required with the realtime CA-Netman interface.

Organization: Sequential card-image file or PDS member.

Space Requirements: Minimal. Should only be large enough to contain schema for those CA-Netman API transactions that are to be issued for CA-7 job completions.

DASD Restrictions: None.

3.2.5.18 CA-Netman API Deferred Request Data Set

Contents: Contains CA-Netman API transactions issued by the realtime CA-Netman interface when the CA-Netman CCI receiver was inactive. Required with the realtime CA-Netman interface.

Organization: DSORG=PS

DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

Space Requirements: The size of this file is dependent on the number of CA-Netman API transactions that may be issued when the CA-Netman CCI receiver is inactive.

DASD Restrictions: None.

3.2.5.19 CA-7 Calendar Partitioned Data Set

Contents: This is an optional data set which is required only if you wish to use online facilities to create and maintain CA-7 base calendars. If used, this PDS contains copies of CA-7 base calendars. While this format differs from the load library versions of CA-7 calendars, CA-7 can use either version to perform schedule resolution.

See member PDSCAL in the CA-7 Sample JCL library (SAMPJCL) for model JCL to allocate a CA-7 calendar PDS.

Organization: Partitioned Data Set (PDS).

DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)
--

Space Requirements: Dependent on the number of calendars used. Ten tracks on a 3380 device should be sufficient space for approximately 200 calendars.

DASD Restrictions: None.

3.2.6 Support File Requirements

3.2.6.1 Log History/Archive Data Sets

Contents: Contains CA-7 log records as they were output by the SAS5 and SAS6 history management jobs. It is recommended that this data set reside only on tape due to the volume of records accumulated.

Organization: These are physical sequential data sets with variable length blocked records.

$DCB=(RECFM=VB,LRECL=2100,BLKSIZE=2100 * n + 4)$
--

Note: n is a user-defined value. Alternately, a BLKSIZE of 32760 can be used for tape data sets.

Space Requirements: If data sets are on tape; none. Otherwise the space requirements are enormous if any extended retention of data is being done. Only user experience can determine the amount needed.

DASD Restrictions: None.

3.2.6.2 CA-7 Cross-Platform Scheduling Profile

Contents: Contains member (CACCENV) which specifies environment variables used by the CA7TOUNI process to submit cross-platform requests from CA-7 to Unicenter TNG or CA-7 Agent on non-MVS platforms. See member XPSPROF in the CA-7 SAMPJCL library for sample JCL to allocate this file.

Organization: DSORG=PO

$DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * n)$
--

Space Requirements: Minimum of one track, two directory blocks.

DASD Restrictions: None.

3.2.6.3 CA-7 Cross-Platform Scheduling Checkpoint

Contents: Contains checkpoint information used by the CA-7 Cross-Platform Tracking System (XTRK) to coordinate tracking functions with non-MVS platforms. See member XPSCKPT in the CA-7 SAMPJCL library for sample JCL to allocate this file.

Note: Each copy of the CA-7 Cross-Platform Tracking System (XTRK) must have its own checkpoint file.

Organization: DSORG=PS

DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096)
--

Space Requirements: One track on a 3390 device will handle checkpointing for 425 remote nodes. One track should be sufficient for most sites.

DASD Restrictions: None.

3.3 ICOM Coupling Facility Structure

If you operate OS/390 in a Parallel Sysplex configuration with a large number of MVS images, ICOM can take advantage of cross-system extended services (XES) to process CA-7 feedback data to reduce DASD contention for the CA-7 Communications data set (COMMDS). XES allows ICOM to manage a structure in a hardware device known as a 'coupling facility'. The coupling facility can be used as a shared I/O device for all participating processors. XES requires any OS/390 release, with appropriate coupling facility hardware and software configuration.

Systems that are candidates for ICOM Coupling Facility exploitation are those where the number of systems where ICOM is running create significant DASD contention for the CA-7 Communications data set. For example, you may have a system with eight OS/390 images. Each image has an ICOM that requires temporary exclusive access to the CA-7 Communications data set to write CA-7 feedback to it. If these systems are in a Sysplex connected through a coupling facility you can use the ICOM Coupling Facility feature to have most of these ICOMs writing their CA-7 feedback to an XES structure in the coupling facility, while one or two of the ICOMs read from the XES structure and write to the CA-7 Communications data set. This can significantly reduce the number of times an ICOM has to wait because the CA-7 Communications data set is 'locked'.

For more specific information on Parallel Sysplex environments and coupling facilities refer to IBM publications *OS/390 Setting Up A Sysplex*, *OS/390 Sysplex Services Guide* and *OS/390 Sysplex Reference Guide*.

3.3.1 ICOM Coupling Facility Structure Lists

An XES structure within the coupling facility consists of one or more data *lists*. There are six lists within the ICOM Coupling Facility structure:

1. CA-7 SMF data for the production (primary) copy of CA-7.
2. CA-7 Trailer data for the production (primary) copy of CA-7.
3. CA-7 NCF data for the production (primary) copy of CA-7.
4. CA-7 SMF data for the test (secondary) copy of CA-7.
5. CA-7 Trailer data for the test (secondary) copy of CA-7.
6. The sixth list is not used at this time.

Each list within the ICOM Coupling Facility structure is treated as a FIFO queue (first-in first-out). New entries are added to the end of the list while existing entries are read from the top of the list. List data elements are 4k (4096 bytes) in size.

3.3.2 ICOM Coupling Facility Structure Implementation

1. Determine if your system is a good candidate for ICOM coupling facility exploitation.
 - a. You must be running an OS/390 (at any level), in a Sysplex configuration with one or more coupling facilities defined.
 - b. You should be running ICOM on at least five MVS images. If you are running fewer images there may be no benefit to using the coupling facility. If you are running too few images you may actually slow the overall throughput by introducing additional overhead.
2. Define the CA-7 ICOM Coupling Facility structure in a CFRM policy using the IBM utility IXCMIAPU. Refer to IBM publication *OS/390 Setting Up a Sysplex* for specific information on this utility. Your input should be similar to this:

```
STRUCTURE NAME(CA7_ICOM_LIST)
  SIZE(size)
  PREFLIST(cf name)
```

Where:

The name of the structure must be CA7_ICOM_LIST.

- | | |
|----------------|---|
| size | The maximum size of the structure, in units of 1K (1024 bytes). For most users, a size of 4 MB should be sufficient 'SIZE(4096)'. See the following discussion to estimate the size needed for your installation. |
| cf name | The name of the coupling facility in which you want the structure defined. This coupling facility must be accessible from all of the images where ICOM executes. |

To calculate the structure size for your installation:

- a. On each OS/390 image where ICOM is running, issue the following MODIFY command:

```
F CA7ICOM,D=STAT
```

This results in the display of two CA-7.575 WTOs with the format:

```
CA-7.575 SMF - aaaa(bbbb), ccccccc(ddddeeee) YES
CA-7.575 TRL - aaaa(bbbb), ccccccc(ddddeeee) YES
```

The 'ddddeeee' values indicate the maximum number of bytes used for this type of record on this image since the last IPL (they are in hexadecimal notation). Convert them to decimal and add the values from the SMF and TRL messages together.

If you are running two copies of ICOM on one or more images, obtain the information from BOTH copies and combine the values.

- b. Add the combined values (SMF & TRL) from the ICOMs on each OS/390 image that will be using the ICOM Coupling Facility feature.

- c. Divide the total by 1,024 to determine the highest threshold in kilobytes (K). Round the number of K up to the next higher value divisible by 1,024. Use this value for the SIZE parameter of the structure definition.

Note: The minimum value you should specify is SIZE(2048). Do not specify a value greater than SIZE(16384) unless you have already experienced problems with that value.

3. Assign coupling facility roles to each ICOM. Each ICOM executes in one of three coupling facility modes. See 6.6.2, "ICOM PARM Values" on page 6-27 for information on mode specification.

- a. Coupling Facility WRITE Mode

WRITE mode causes ICOM to write the SMF, Trailer, and NCF data it collects to the ICOM Coupling Facility structure. Data elements are added to the end of the appropriate list in the structure.

- b. Coupling Facility READ Mode

READ mode causes ICOM to read data from the ICOM Coupling Facility structure as well as collecting SMF, Trailer and NCF data from the local environment. All data is written to the CA-7 communications data set or the NCF communications data set. Coupling Facility entries are read from the top of each list using a read/delete function.

- c. No Coupling Facility Interface

Not all ICOMs are required to interface with the Coupling Facility. For example, you may have one or more MVS images which are outside of the SYSPLEX but still have shared access to the DASD where the CA-7 Communications data set resides.

You should use the following guidelines to assign Coupling Facility roles to each ICOM:

- a. You should have at least 2 ICOMs running in READ mode. This ensures that even if one of these ICOMs, or the image it is running on, is shut down, the feedback being written to the Coupling Facility is still being read and transferred to the CA-7 Communications data set.
- b. You should have at least 1 ICOM running in READ mode for each three ICOMs that are running in WRITE mode. Otherwise, you may have a situation where the one or two READ ICOMs become a bottleneck attempting to transfer large amounts of data from the Coupling Facility to the CA-7 Communications data set.
- c. You should assign READ modes to the ICOMs on those images which process the least amount of CA-7 related work. This will help 'spread' the processing load to ICOMs which have lighter local workloads.

3.4 Calendar Definition and Structure

Base calendars are used by CA-7 to determine available dates for job and input network scheduling for schedules defined on a date/time basis.

The base calendar definitions within CA-7 allow for processing schedule variations. Different types of calendars may be required depending on the type of processing being done. Fiscal year calendars, billing calendars, workday calendars, and payroll calendars are a small sample of the types of calendars which may be required by the user in scheduling the production workload.

CA-7 allows the user to define as many different calendars as necessary, up to a maximum of 1500.

CA-7 calendars can be generated using batch or online facilities. The process to generate calendars through batch facilities is described in the following pages. Online calendar maintenance requires the allocation and identification of a CA-7 calendar PDS. See the 5.2.23, "CALENDAR Statement" on page 5-56 and the Online Calendar Maintenance screen (DB.2.8) in the *CA-7 Database Maintenance Guide* for detailed information about maintaining calendars through online facilities.

Batch generation of base calendars can be accomplished with the CALENDAR macro. The user must identify all calendars required for scheduling the existing environment. These calendars must then be defined by CALENDAR macro parameter values. Base calendars are then generated by assembling and linking the calendar definitions to reflect the format specified by these parameters. These generated calendars are stored in the CA-7 load library or a user-specified calendar load library that is concatenated to the STEPLIB.

If a CALENDAR statement is not included in the CA-7 initialization file, a CALBLK statement must be present with NAME= being the name of the load module that is linked into the load or Calendar library in order for this calendar to be available for job scheduling.

The flexibility of the combined calendar/schedule concept allows the user to define and/or change calendars well in advance of their actual use. For example, all calendars to be used the following year can be generated and stored in the load library at any time and then referenced when actually required.

CA-7 also includes facilities for listing (or displaying) calendars for review or reference purposes. An actual calendar format is printed which reflects each month's beginning and ending points and all scheduled workdays (nonworkdays are removed). Following each calendar, specified holidays are listed for reference. The following is a partial base calendar assembly listing.

Sample Base Calendar Assembly Listing

```

129+*,*****
130+*,**
131+*,**          CA-7 BASE CALENDAR SCAL94WD
132+*,**          FOR YEAR 1994
133+*,**          DATE 93064
134+*,**
135+*,*****

137+*,*****
138+*,**
139+*,**  MONTH 1          JAN
140+*,**
141+*,**          SUN  MON  TUE  WED  THU  FRI  SAT
142+*,**
143+*,**
144+*,**          3    4    5    6    7          (B1)
145+*,**          10   11   12   13   14
146+*,**          17   18   19   20   21
147+*,**          24   25   26   27   28
148+*,**          (E1)
149+*,**
150+*,*****

152+*,*****
153+*,**
154+*,**  MONTH 2          FEB
155+*,**
156+*,**          SUN  MON  TUE  WED  THU  FRI  SAT
157+*,**
158+*,**          (B2)  2    3    4
159+*,**          7    8    9   10   11
160+*,**          14   15   16   17   18
161+*,**          21   22   23   24   25
162+*,**          (E2)
163+*,**
164+*,*****

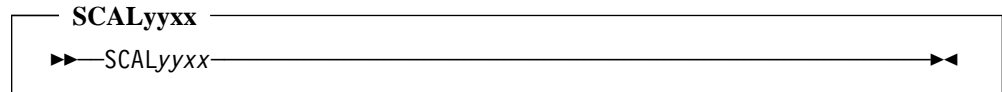
166+*,*****
167+*,**
168+*,**  MONTH 3          MAR
169+*,**
170+*,**          SUN  MON  TUE  WED  THU  FRI  SAT
171+*,**
172+*,**          (B3)  2    3    4
173+*,**          7    8    9   10   11
174+*,**          14   15   16   17   18
175+*,**          21   22   23   24   25
176+*,**          28   29   30   (E3)
177+*,**
178+*,*****

```

3.4.1 CALENDAR Macro

A data center's base calendars are generated by assembling and link editing CALENDAR macros with the user-supplied values. One CALENDAR macro is used to create each module which defines each base calendar. Calendars are used by the schedule resolution application.

A standard naming convention has been established for referencing base calendars. This format must be adhered to in order to guarantee that all calendar names are referenced correctly. The required format is as follows:



Where:

SCAL

Is constant.

yy

Must be the year the base calendar is defining.

xx

Are two nonblank characters supplied by the user to ensure calendar uniqueness.

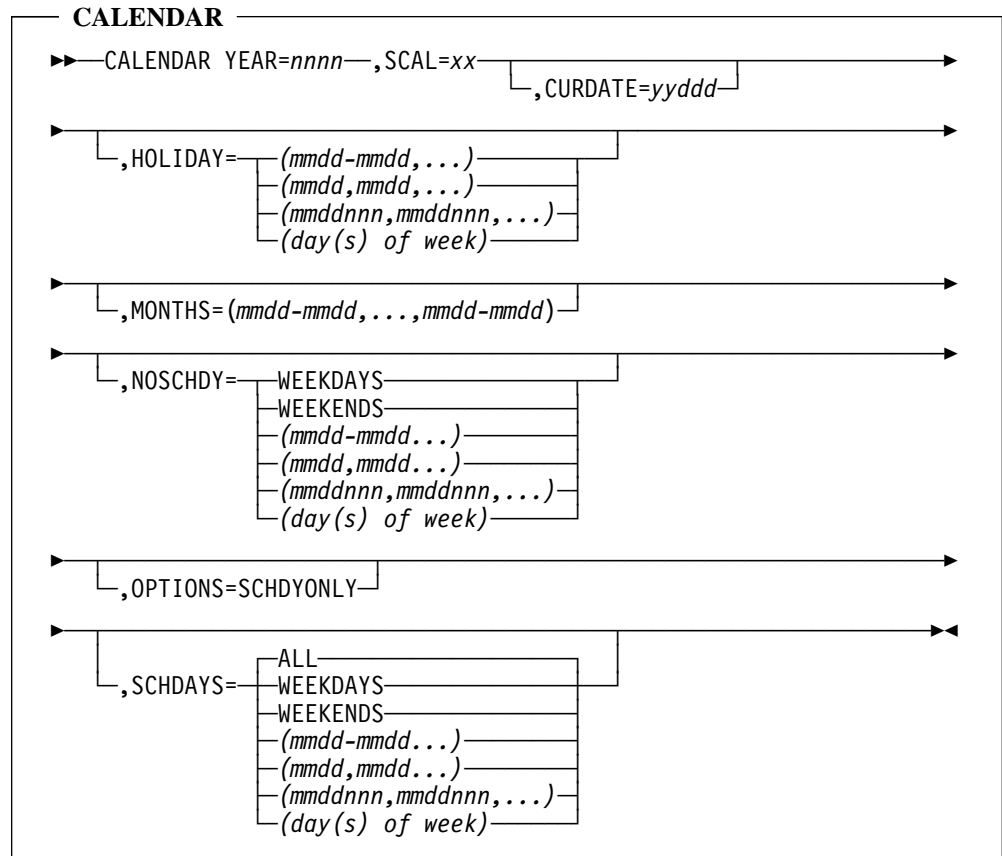
For example, SCAL01WD is the name of a base calendar for the year 2001 with the unique characters WD (which might identify a weekday calendar).

Prior to beginning the calendar generation process, the user should establish standards and procedures to control the assignment of the last two unique characters. This ensures consistency and guarantees duplicate names are not chosen.

Base calendars must be link edited using this standard naming convention (SCALyyxx).

Some sample calendars are generated by the CA-7 SYSGEN.

3.4.1.1 Syntax



Where:

YEAR

Specifies the year being defined by this base calendar. It supplies the yy segment of the calendar name. The value is specified as the 4 numeric characters defining the year (for example, 2000). A required parameter.

SCAL

Specifies the unique characters which form the xx segments of the calendar name. The value must be 2 nonblank characters, either alpha, numeric or both. A required parameter.

CURDATE

Specifies the date when this base calendar was assembled. This information is stored in the calendar record for documentation purposes. The value must be in Julian date form. This keyword is optional.

HOLIDAY

Specifies which days in this calendar are holidays. Designation of holidays by the HOLIDAY parameter is purely documentary since holidays do not halt production. If a holiday is not an available processing day, it must be specified by NOSCHDY. An optional parameter. For parameter definitions see SCHDAYS.

MONTHS

Specifies the beginning and ending days for each of the 12 months (January through December). It must be used if the calendar being defined has nonstandard months; for example, a fiscal or accounting calendar. The value must specify the desired beginning and ending days of each month as follows. (See 3.4.2, “Coding Notes” on page 3-35.) An optional parameter.

mm

Identifies the month number.

dd

Identifies the day of the month.

NOSCHDY

Specifies which days of the year are not to be designated as available processing days. An optional parameter. For parameter definitions see *SCHDAYS*.

OPTIONS

Defines calendars being referenced by schedules using relative number of processing days from the beginning or end of the month to determine processing cycle. The value must be specified as *SCHDYONLY* if used. This value causes CA-7 to count only available processing days when resolving a schedule with number of days relative to the beginning or end of the month. (See the *RDAY* field on the DB.2.1 screen in the *CA-7 Database Maintenance Guide*.) The result of not specifying *OPTIONS=SCHDYONLY* is that all days are counted when resolving number of days relative to the beginning or end of the month. An optional parameter.

SCHDAYS

Specifies which days of the year are to be designated as available processing days.

ALL

Indicates every day is to be considered an available processing day. This is the default value.

WEEKDAYS

Indicates only Monday through Friday in each week are available processing days.

WEEKENDS

Indicates only Saturday and Sunday are available processing days.

mmdd-mmdd,...

Specifies a range of days by month and day.

mmdd,mmdd,...

Specifies individual days by month and day.

mmddnnn,mmddnnn,...

Specifies a start day, by month and day, and a number of days (nnn) beginning with that day to be considered.

day(s) of week

Specifies one or more of SUN, MON, TUE, WED, THU, FRI, or SAT.

3.4.2 Coding Notes

- The macro name CALENDAR should begin in column 10 and there must be at least one blank between the macro name and the first parameter.
- Continuation of the CALENDAR macro statement is indicated by a nonblank character in column 72. Macro parameters and values may be coded through column 71 with no blanks. All continuation statements must begin in column 16. Keyword values may be broken for continuation at a comma. This is a standard assembler continuation.
- The CALENDAR macro parameters are keyword type and nonpositional. They may be specified in any order.
- If coding nonstandard months (fiscal year), a year-end boundary may only be crossed in the first or twelfth month (that is, the first month may begin in December, January or February). The last month may end in November, December or January, but it must not start in January. Also, twelve months must be specified.
- If the beginning day for a month is more than one month greater than the beginning day coded for the previous month, the assembly of the calendar does not show the ending day for the previous month.

For example, if MONTHS=(0101-0301,0302-0401,...) then the assembly shows a (B1) for January 1, a (B2) for March 2, but the (E1) for March 1 does not appear on the calendar from the assembly.

3.5 Notification of CA-7 Job Completions Using CAIENF

CAIENF events are created for CA-7 job completions. Applications running outside the CA-7 address space may be notified through CAIENF of the completion of CA-7 submitted jobs.

These events are not recognized by CAIENF however unless the CAL2DCM0 module is added to the CAIENF database. The CAL2DCM0 module describes the CA-7 job completion event for CAIENF. To add this module to the CAIENF database, see the L2DCM0 member on the CA-7 SAMPJCL library.

Tailor the JCL for required job statement changes and also specify correctly the data sets indicated in the comments. Two SYSIN statements are provided. As the comments there indicate, one is used if the ENF task is active and the other is used when the ENF task is inactive. Remove the comment from the appropriate statement and run the job. All STEPLIB libraries in this job must be APF-authorized.

When this process is complete, the CA-7 job completion event is identified to CAIENF as the L2JOBCMP event. See the Unicenter TNG Framework for OS/390 *Administrator Guide* for information on displaying the contents of ENF event records such as L2JOBCMP.

3.6 Notification of CA-7 Browse Messages Using CAIENF

CAIENF events can be generated for messages sent to the CA-7 browse data set. Applications running outside of the CA-7 address space may be notified of these messages through CAIENF. These events are not generated by CAIENF unless the L2DCM1 module is added to the CAIENF database. The CAL2DCM1 module describes the CA-7 browse message event (CA7BRWSE) and the message text data element (CA7TEXT) to CAIENF. To add this module to the CAIENF database, see member L2DCM1 in the CA-7 Sample JCL Library (SAMPJCL). See the *Unicenter TNG Framework for OS/390 Administrator Guide* for information on displaying the contents of ENF event records. The CA-7 browse event is used by some other CA products such as CA-Opera and CA-7/Smart Console and may have already been added to your CAIENF database.

3.7 Workload Balancing

The workload balancing facility analyzes jobs waiting for execution and sets priorities for submittal by preset criteria. The preset criteria include:

- start times
- CPU usage
- tape drive usage
- user-assigned priorities
- external events

When the facility is active and objective criteria have been defined, CA-7 analyzes every job waiting for execution whenever it is ready for processing or completes processing. As resources are made available or released back to the operating system, the defined pools are adjusted automatically to reflect current availability.

Each job waiting for processing is analyzed, its priority rewarded or penalized based on every defined value. It is then automatically submitted whenever the priority reaches the defined threshold value or becomes the highest of all jobs waiting for submittal. The job with the highest priority that can run within the available resources is always the next one submitted. If priorities are equal, the number of available initiators may cause multiple smaller jobs to be submitted rather than one large job, so that maximum use of initiators can be attained.

The base priority for a job can be entered on the DB.1 screen when initially defined to CA-7. The priority can be any value from 1 to 255. Priority 255 is reserved to force a job to be processed as soon as all prerequisites have been satisfied. It is called the express priority. CA-7 provides a default priority value of 100. Whenever a job exceeds the maximum allowable quantity of resources, the maximum penalty defined is assessed by subtracting that value from the priority. If a job requires the minimum for any processing objective, the maximum reward is added to the job priority. For requirements between the minimum and the maximum, priority is adjusted by a percentage of the reward. The percentage is determined by the relationship of the requirement to the difference between the minimum and maximum. For example, if the range is five units and the requirement is three, 60 percent of the reward is given.

All of these decisions are made continuously based on the review of current conditions. Priority values may increase or decrease whenever another event causes processing requirements or availability to change. Relationships to events or time of day can cause priorities to change.

Note: VRM Device Control provides an alternative to Workload Balancing's control of job submission based on tape drive availability. When this VRM option is activated, VRM resources representing tape and/or disk storage devices will be dynamically defined as part of the CA-7 database load process.

This option should be considered if the limit on the number of Workload Balancing tape drive types is too restrictive. Where Workload Balancing allows for only two tape drive types, the number of device groupings with VRM Device Definition is determined by the user.

Also note that VRM Device Control operates independently of Workload Balancing and does NOT affect WLB priority calculations.

See Chapter 5 of the *CA-7 Database Maintenance Guide* for an extended discussion of this option.

3.7.1 Defining the Production Environment

Processing objectives involve far more than making a simple choice. A combination of different considerations is necessary. To start with, the following elements are described:

- Tape drives
- CPU use
- Initiators or job class structure
- Job start times
- Threshold priorities.

A workload balancing questionnaire and its directions are provided in 3.7.4, “Workload Balancing Questionnaire” on page 3-45. This questionnaire can assist in establishing a startup objectives definition. By answering the questions as they apply to your data center operation and following the questionnaire directions, a sample source module can be generated. The sample source module can then be assembled, link edited and DEMANDED through CA-7 to begin using workload balancing for realtime selection of jobs for submission. The workload planning facility can also be helpful in fine tuning the objectives definition.

3.7.1.1 Tape Drives

Tape drives play an important role in determining the completion time of a job. With workload balancing in effect, CA-7 submits a job only when enough tape drives are available and only submits as many jobs as the available tape drives allow. The tape drives may be of various types: 6250 BPI, 1600 BPI, 800 BPI, 7-track, 9-track, and so forth. CA-7 currently controls only two types at any one time. The user defines what these two types are with the TAPE1 and TAPE2 workload balancing macros and the SASSUTBL module. One particular group of tape drives, based on the IBM device codes, is considered TYPE1 and the other as TYPE2. (See 3.7.5, “Defining the Types of Tape Drives” on page 3-53.) The default is TYPE1, so if device code is not defined under either type, it is considered TYPE1.

The following are the objectives in controlling tape drives:

- To use as many tape drives and initiators as possible at the same time without going over the maximum allowable number of either. For example, if two initiators and three tape drives are available, the objective is to submit two jobs. One job uses one tape drive and the other uses two tape drives (assuming such jobs are available for submission), rather than one job alone using three tape drives.
- To limit the number of jobs submitted so that the total number of tape drives needed does not exceed the maximum number allowed to be scheduled, regardless of the resultant priority. For example, if the number of tape drives CA-7 is allowed to schedule is ten, and the number of tape drives currently being used is eight, then CA-7 does not submit any job(s) using more than two tape drives, even if such job(s) have high priority.
- To limit the jobs using tape drives outside the allowable limits; that is, less than the minimum and more than the maximum allowable. For example, during a time when tape drive usage is to be completely discouraged, maximum tape drives allowed per job can be set to zero so that no jobs needing tape drives are scheduled. When jobs having heavy tape drive usage requirements are to be encouraged, a minimum allowable number can be set to a relatively high value so that a job using less tape drives than the minimum value is delayed from being submitted.
- To give a boost to a job that needs a large number of tape drives and is defined as difficult to schedule. For example, two initiators and seven tape drives are available, and a choice is to be made among three jobs; Job A requiring three tape drives, Job B requiring four and Job C requiring seven. Normally, Job A and Job B are given higher rewards since all tape drives and initiators are used, if both jobs are submitted. Job C is held back even though enough tape drives and initiators are available. If having seven tape drives is considered difficult, difficult to schedule (DTS) can be defined as seven and Job C gets an additional boost, whereas, Job A and Job B do not.

3.7.1.2 CPU Use

Another objective of workload balancing is to balance the CPU cycle use. If the jobs present in the CPU are process bound jobs, CA-7 tries to submit I/O bound jobs (and vice versa) to ensure a balanced mix of jobs.

The main objective for balancing CPU use is to achieve an ideal CPU use per job.

CPU use is considered the ratio of CPU time to the elapsed time for a job expressed as a percentage. For example, if a job needs six seconds of actual CPU time in one minute of elapsed time, then the job is said to have a CPU use of 10 percent. A simple average of CPU use figures for all jobs active on the CPU, or actual CPU use per job, is taken. If actual CPU use per job is less than the ideal use per job, then the jobs with CPU use higher than actual use are given rewards. For example, if ideal use per job is set at 10 percent, and actual is 8 percent, then all jobs with CPU use higher than 8 percent are given rewards and jobs less than 8 percent are given penalties. Jobs bringing actual use closer to ideal are given higher rewards than others.

3.7.1.3 Initiators and Job Class Structure

The third objective of CA-7 is to consider the number of initiators available and the job class structure. CA-7 fills as many initiators as possible without going over the allowable number. In doing so, CA-7 again makes sure not to exceed the maximum number of allowable jobs in each category or class of jobs. (The class code is identical in appearance to OS job classes, but is only used internally within CA-7 and does not have to match the initiator structure or JOB statement class specification.) For example, if a total of 10 initiators are allocated to jobs running under CA-7, and eight jobs are already running, then CA-7 submits only two more jobs to the CPU. Again, CA-7 selects these two jobs from the classes whose maximum number running under them has not been reached. It is then possible to control the total number of jobs running under CA-7 and the number of jobs from each class.

3.7.1.4 Job Start Times

The next objective of CA-7 is to consider how late or early the job is according to a due-out time and the time available to correct the situation. For example, if two jobs, Job A and Job B, are both five minutes late, and Job A runs for 10 hours while Job B runs for 10 minutes, then CA-7 tries to submit Job B ahead of Job A. If both are five minutes early, then CA-7 tries to submit Job A ahead of Job B.

If a job is late, it is rewarded. If it is early, it is penalized. A job is not considered late (or early) until it is late (or early) by a user-defined factor (RUNTF) of its elapsed time. For example, if RUNTF is defined as 10, then a job which runs for an hour is not rewarded until it is late by six minutes or more (60 minutes multiplied by .10) or a job which runs for 10 minutes is not rewarded until it is late by one minute or more (10 minutes multiplied by .10). Among jobs which are late by an equal amount of time, shorter running jobs are given more reward than longer running jobs. Among jobs which are early by an equal amount of time, longer running jobs are given less penalty than shorter running jobs.

3.7.1.5 Threshold Priorities

By using the previous features, the original job priority is changed by either adding rewards or subtracting penalties. The resultant priority is checked against the threshold priority. Unless the number of jobs running on a CPU is less than the specified minimum jobs that must run, the resultant job priority must be higher than the threshold priority. There must not be any resource constraints in order for CA-7 to submit that job to a CPU. The objective is clearly the best mix of jobs that can be submitted. A job may get the highest reward in one category, such as tape drives, but more penalties in others and the resultant priority is not high enough for the job to be submitted to the CPU. That job is held back and others with higher priorities are submitted. This selection of jobs is clearly beneficial in a situation where work is backlogged due to downtime.

In order not to hinder submission of jobs that are of special interest, a special job priority and special classes for jobs have been provided.

A priority of 255 has been reserved as a special job priority. If a job is given an original priority of 255, it indicates that the job is important and the relative importance of this job with respect to others need not be checked. If there are no resource constraints (for example, no tape drives available or no initiators available), the job should be submitted.

In addition, up to three different special job classes can be defined with their own threshold priorities. The class of the job that needs to be submitted ahead of others should be changed to one of these special CA-7 classes. For example, the threshold priority is set to 200 and class B has been defined as a special class with a threshold priority of 50. If two jobs exist, JOBX in class C with a resultant priority of 150 and JOBY in class B with a resultant priority of 60, then JOBY is submitted ahead of JOBX, since JOBY belongs to a special class and has a priority higher than its threshold value.

3.7.2 Implementing WLB

Implementing the workload balancing facility (WLB) is a matter of getting a processing objective in the form of a load module into the CA-7 system so that CA-7 balances the workload to meet the processing objectives of the data center.

In reality, most shops need more than one processing objective to properly balance the workload. For example, during business hours, one or more online systems may be in use; whereas, after business hours the online systems are not needed and so are shut down, freeing up system resources that can be used for additional batch processing. In this case, at least two resource pictures are needed for WLB to work effectively.

The general steps for implementing WLB for each different processing objective are shown in the following table.

Table 3-2. Implementing WLB	
Step	Action
1	Analyze the needs and requirements of the environment. A WLB questionnaire is provided in 3.7.4, “Workload Balancing Questionnaire” on page 3-45 to aid the user in analysis of the environment, determining the needs, and coding the macros.
2	Define the types of tape drives. The system permits two logical classes of tape drives.
3	<p>Code the WLB macros. There are six macros which cover the basic resources of the environment, and two format macros:</p> <p>WLBPDEF Module name, required for format</p> <p>TAPE1 Class of tape drives</p> <p>TAPE2 Class of tape drives</p> <p>CPU CPU use</p> <p>INTR Initiator characteristics</p> <p>CLBARR Job class barriers</p> <p>STARTIME Scheduled start time for jobs</p> <p>WLBPEND Required for format</p>
4	Assemble and link edit the macros creating the module containing the processing objective in the LOADLIB. The name for this module must begin with the characters UCC7R.
5	<p>Add a job to the database that has a job name that matches the module name created in Step 4. To add a job, proceed as follows:</p> <ol style="list-style-type: none"> Bring up the DB.1 screen. Enter ADD to select the ADD function. Enter the module name used in WLBPDEF in the JOBNAME field. <p>No other fields are necessary for the definition. Any job whose name begins with UCC7R is automatically treated as a nonexecutable job by the system. Whenever the job enters the queues, only the processing objective criteria is changed.</p>
6	Define a schedule or a trigger for the job added to the database in Step 5. The job scheduling for a WLB module/job is the same as for any other job. This avoids having to manually DEMAND schedule the processing objective when it is needed.

3.7.3 Defining the Selection Parameters

To use the workload balancing facility, the user must supply the processing objectives criteria by defining various parameters to CA-7. The criteria defines a virtual configuration for CA-7 to manage. This virtual configuration does not have to match the actual environment. It may be desirable to omit some resources that are not to be used for CA-7 jobs. The objectives criteria also let CA-7 know how much reward or penalty to assign to a job under consideration and under what conditions.

Many different configurations and processing objectives may exist. Definitions for different shifts, weekdays and weekends are just a small sample of the types of parameter definitions which may be desired. CA-7 allows the user to define as many different definitions as necessary. However, only one definition is used at any given time.

CA-7 generates these definitions based on the parameters defined to the macros specifically provided for this purpose. These generated definitions become nonexecutable load modules and jobs which are stored in the CA-7 load library and database. The names of these load modules must be names of the format UCC7Rxxx where xxx can be any alphanumeric characters. UCC7RDFL is a reserved name; a default definition under that name is included on the version tape.

The scheduling feature of CA-7 can be used to make different definitions effective only at specific times. With every workload balancing parameter definition, a nonexecutable job with the same name as the load module must be added to the database. This generic group of jobs with names such as UCC7Rxxx can have prose, schedules, and triggers, just like jobs in the database. These jobs can be demanded at anytime. However, they are never submitted to the CPU or executed in the initiators. When in the request queue, they are intercepted by CA-7 and a definition (load module) with the same name is loaded. (For example, if job UCC7R010 is demanded, a UCC7R010 definition is be loaded.) Whenever CA-7 is active, some definition must be present. In the absence of another definition, the default definition, UCC7RDFL, provided with the version tape, is used.

Note: If the default definition is used, workload balancing is deactivated.

3.7.4 Workload Balancing Questionnaire

Following is a questionnaire which can help in creating a starter objectives definition. By answering the questions as they apply to your shop and following the workload balancing questionnaire directions which follow, a sample source module can be generated. That module can then be assembled, link edited and DEMANDED through CA-7 to begin using workload balancing for the online selection of jobs for submission. Users find the workload planning facility helpful in fine tuning their objectives definition.

Answers to the following questions should relate to one particular time period (shift) which has fairly consistent processing characteristics.

The following questions relate to tape drive use. If your shop uses two different types of tape drives (for example, reels and cartridges), answer each question twice, once for each type of tape drive.

Tape Drive Usage

TYPE 1 TYPE 2

1. How many physical tape drives are available for CA-7 jobs during this time period? _____

If the answer to #1 is 0, skip to Initiators.

2. What is the average number of drives required per job (average of "high-water mark" for each job)? Take into consideration the number of jobs which do not require tape drives. May be a fractional value. _____
3. What is the maximum number of tapes that you would expect any single job to use? _____

Initiators

4. How many initiators are available for job submission from CA-7 during this time frame?
5. Of those initiators available, how many would you expect to be occupied by CA-7 jobs at any particular time?
6. What is the total maximum percent of CPU use that CA-7 submitted jobs should occupy? (Consider leaving machine time available for other online systems, test jobs, and system overhead.)
7. Are there jobs to be run during this time frame which have significantly higher priority than the average job?
8. Are there jobs which can be run at this time, but which have such a low priority that you would like to see them run only if there is little else in the system?

Class Barriers

9. Is there a set of jobs which should not run concurrently because of file requirements, resource conflicts, or any other special needs?
10. Are there any jobs to be run at this time that are backup to an online database and which should not be run concurrently with the online system?
11. Do you have multiple users who share your system and for whom you wish to provide varying levels of service?
12. Will you be using the LOAD command to build database profiles? If yes, Class 8 must have nonzero count.
13. Will the RUN command be used to schedule certain request type jobs? If yes, Class 9 must have nonzero count.

Timely Job Completion

14. At a certain point a job is very late and if it is a few more hours late, it is not more delinquent.

Example: Job A is late by 12 hours. If it does not run for three more hours, it would still be considered a half day late. At what point in your shop would you quit counting how many hours late a job is and just consider it very late? Maximum of 12 hours.
15. At what point would you stop counting how many hours early a job is and just consider it very early? Maximum of 12 hours.

Resource Priority

Rate the following statements according to relevance to your shop. Enter a rating from 0 to 10, 0 being not a concern at all, 10 being extremely important in your environment.

16. It is important to your shop to balance the initiators with a good mix of CPU bound and I/O bound jobs.
17. If a job is early, it should not be run until closer to its start time since it may use resources required by a subsequent job which is on time.
18. If a job is late it should run as soon as possible, even at the possible expense of some wasted resources.

TYPE 1 TYPE 2

19. It is important not to overschedule tape drives, possibly causing a job to wait in an initiator (or JES3 queue) until tape units can be allocated.
20. It is important to maximize use of available tape drives.

_____	_____
_____	_____

3.7.4.1 Workload Balancing Questionnaire Directions

The term #nn in the following instructions for creating a workload balancing (WLB) definition module refers to the question numbers on the preceding questionnaire. The calculated values can be entered in the blanks with corresponding labels on the sample workload balancing objectives definition in Workload Balancing Objectives Definition on page 3-51.

3.7.4.2 WLBPDEF

MODNAME

Enter the last 3 characters of the UCC7Rxxx module name to be used for this definition.

VRSN

Enter the version number.

3.7.4.3 TAPE1/TAPE2

NAME

This entry may be up to 8 characters in length and is informational only.

TOTAV

Enter the value given in question #1.

MXREW

Enter the answer to question #18 multiplied by 3 ($\text{MXREW} = \#18 \times 3$).

MXPEN

Enter the answer to question #17 multiplied by 3 ($\text{MXPEN} = \#17 \times 3$).

MNDTS

Enter a value over halfway between #2 (average drives/job) and #3 (maximum number of drives required/job).

$$(\text{MNDTS} = ((\#3 - \#2)/2) + \#2)$$

MXDTS

Enter the value given for #3 (maximum drives/job).

MXBST

Tape rewards tend to be higher for jobs requiring smaller numbers of drives to use both tapes and initiators. If a large number of drives is available, you may want to boost a job requiring a large number of drives. Ideally, the larger #3 (maximum drives/job) is, the larger MXBST should be. It should also be related to MXREW since that value reflects the relative importance of tape scheduling in your shop. Also, the more overscheduled your tapes drives tend to be ($\#5 \times \#2$) as compared to #1), the higher you would want to boost. The following calculation takes these factors into account in figuring a MXBST value. Start with this value. If it is not generally enough to get a large tape job scheduled when the right number of drives is available, you may want to increase it.

$$(\text{MXBST} = (\text{MXDTS} + ((\#5 - 1) \times \#2)) \times \text{MXREW} / \#1)$$

MNJAL

MNJAL and MXJAL are most useful as realtime adjustments to the WLB definition. MNJAL may be raised temporarily, using the XWLB screen, in a situation where none of the large tape jobs can break into the system because small tape jobs are taking up the slack. Begin your definition with MNJAL=0.

MXJAL

Enter the answer to #3. If large tape jobs are dominating all drives and jobs requiring few drives are not moving through the system at a high enough rate, lower MXJAL temporarily using the XWLB screen.

Note: Jobs with tape requirements falling below MNJAL or above MXJAL are not scheduled at all. The system disregards rewards and penalties related to lateness, CPU use or difficult-to-schedule tape requirements for such jobs.

MXTAL

This value allows for overscheduling of physical tape drives to take up the slack on the system created by variances between job step requirements for tape drives. Begin with the values suggested below. If tape jobs are consistently waiting in the system initiators for tape drive allocation, lower MXTAL ($MXTAL = (1.25 \times \#1)$ or $(\#4 \times \#2)$, whichever is smaller).

3.7.4.4 CPU

IDLUT

Divide the answer to #6 by #4 to give ideal CPU use per CA-7 job. If the value of #4 is significantly larger than #5, use the value of #5 instead of #4 ($IDLUT = \#6 / \#4$).

MXREW

Multiply #16 by 3 for maximum reward for ideal CPU use ($MXREW = \#16 \times 3$).

MPEN

Use the same value as MXREW ($MPEN = MXREW$).

3.7.4.5 INTR

TOTAV

Enter the answer to #4.

MNJOB

The purpose of MNJOB is to discourage idle hardware and CA-7 withholding jobs from submission because they do not meet the JPTHR value specified. When the number of CA-7 jobs on the system falls below this value, the relative priority of jobs ready to submit is ignored. They are submitted on a first come first serve basis (within physical restraints such as tape drive availability) until the number of jobs on the system exceeds MNJOB. To start, use ($MNJOB = \#4 / 2$).

This value is obtained by observing machine idleness versus CA-7 ready queue backlog in your production environment. Begin with the value calculated above and adjust it according to your shop's performance.

JPTHR

The priority derived by WLB reward/penalty calculations should reflect that job's impact on the mix of jobs currently running on the CPU. A job's final priority is the sum of the WLB rewards and penalties and the job's original priority, assigned on the DB.1 screen. JPTHR is the threshold of priority below which an average job is considered "too bad" for the current job mix to submit unless MNJOB is not exceeded. A value of 75 is recommended for an initial value since the average job enters the queues with the default priority of 100. It must incur a net penalty from WLB of 25 or more to fall below this value (JPTHR=75).

SPCLS1/SPCLS2/SPCLS3

If the answer to #7 is yes, you might want to categorize jobs of higher priority into a class (not necessarily related to job class) designated by A through Z, 0 through 9 on the DB.1 screen. You should also give the job a higher job priority on the DB.1 screen. The average job receives a default priority of 100. The default or user-specified job priority is added as a WLB reward as the jobs enter the queues. JPTHR is set at 25 percent below the original priority of the average job. For SPCLSx jobs of higher priority, you may want an even lower threshold specified as SPCLSx=(Y,0) where x is 1, 2 or 3, Y is the class assigned for that particular group of jobs and 0 is that class's threshold priority. Keep in mind that the default class for those jobs considered average is A.

If the answer to #8 is yes, you can enter a lower than average job priority on the DB.1 screen and specify SPCLSx=(Y,150). This is to ensure, for example, that the job is submitted only if it has received enough rewards from WLB for CPU and tape use so that it is considered good for the current job mix on the system. Or to ensure that if it became so late, the reward would push it over its threshold.

3.7.4.6 CLBARR

BARx

If the answer to #9 is yes, you can put those jobs into the same class on the DB.1 screen and set a class barrier of 1 to single-thread those jobs through the system. For example, if several jobs which impact a particular disk pack heavily are given a class C on their DB.1 screens by specifying BARC=1 to WLB, only one of those jobs runs at a time.

If the answer to #10 is yes, you can overcome a scheduling problem using class barriers in combination with scheduling of different WLB definitions. Assume that your online system (which may or may not be under CA-7 control) has a scheduled downtime from 1:00 AM to 5:00 AM daily for backup purposes. You have a backup program which runs under CA-7 with a class D on the DB.1 screen and has an elapsed time of three hours. If the backup does not enter the system by 2:00 AM it cannot be run until tomorrow's downtime. A WLB definition module specifying BARD=1 can be scheduled for 1:00 AM and a separate WLB definition module specifying BARD=0 can be scheduled for 2:00 AM daily. This allows the backup program to enter the system only between 1:00 and 2:00 AM.

If the answer to #11 is yes, you may want to classify each user by class on the DB.1 screen and control the amount of access each class receives. For example, user 1 represents a large portion of your site's revenue, so you set user 1's class barrier higher than that of user 2, who accounts for a small percentage of your business, but whose jobs may come in sporadically.

3.7.4.7 STARTIME

MXLAT

Enter the value assigned to #12 for the point past which a job receives no more than the maximum reward for being late.

MXERL

Enter the value assigned to #13 for the point before which a job receives no more than the maximum penalty for being early.

MXREW

Enter the value in #18 multiplied by 3 for the maximum reward a job receives for being late ($MXREW = \#18 \times 3$).

MPEN

Enter the value in #17 multiplied by 3 for the maximum penalty to be assigned to a job for being early ($MPEN = \#17 \times 3$).

RUNTF

Runtime factor is the percentage of job elapsed time which is used as leeway in calculating whether a job is late. This leeway allows a job which runs for 10 minutes, and which is 10 minutes late, to be run ahead of a job which runs two hours and which is also 10 minutes late. The first job receives a reward for lateness, while the second job does not. RUNTF=10 is recommended and is the default. If varying this value seems to help overall job timeliness in your installation, it can be manipulated.

Sample Workload Balancing Objectives Definition

	PRINT NOGEN		
	WLBPDF MODNAME=_____	3 characters, no default	X
	VRSN=_____	3 digits, informational	
*	TAPE1 NAME=_____	8 characters, informational	X
	TOTAV=_____	0-255, default=0	X
	MXREW=_____	0-255, default=20	X
	MXPEN=_____	0-255, default=20	X
	MNDTS=_____	0-255, < MXDTS, default=0	X
	MXDTS=_____	0-255, > MNDTS, default=0	X
	MXBST=_____	0-255, default=0	X
	MNJAL=_____	0-255, < MXJAL, default=0	X
	MXJAL=_____	0-255, > MNJAL, default=0	X
	MXTAL=_____	0-255, not < TOTAV, default=12	
*	TAPE2 NAME=_____	8 characters, informational only	X
	TOTAV=_____	0-255, default=0	X
	MXREW=_____	0-255, default=20	X
	MXPEN=_____	0-255, default=20	X
	MNDTS=_____	0-255, < MXDTS, default=0	X
	MXDTS=_____	0-255, > MNDTS, default=0	X
	MXBST=_____	0-255, default=0	X
	MNJAL=_____	0-255, < MXJAL, default=0	X

	MXJAL=_____	0-255, > MNJAL, default=255	X
*	MX TAL=_____	0-255, not < TOTAV, default=12	
	CPU IDLUT=_____	0-100, default=8	X
	MXREW=_____	0-255, default=20	X
*	MXPEN=_____	0-255, default=20	
	INITR TOTAV=_____	0-255, default=10	X
	MNJOB=_____	0-255, default=255	X
	JPTHR=_____	0-255, default=100	X
	SPCLS1=_____	(x,yyy), default (yyy) = 100	X
	SPCLS2=_____	same as SPCLS1	X
	SPCLS3=_____	same as SPCLS1	X
*	CLBARR BARA=_____	0-255, default=255	X
	BARB=_____	same default=255	X
	BARC=_____	same default=255	X
	etc...		
	BAR9=_____	same default=255	
*	STARTIME MXLAT=_____	0-12, default=12	X
	MXERL=_____	0-12, default=12	X
	MXREW=_____	0-255, default=20	X
	MXPEN=_____	0-255, default=20	X
	RUNTF=_____	0-100, default=10	
*	WLBPEND		
	END		

3.7.5 Defining the Types of Tape Drives

The module SASSUTBL defines a table of units and device codes. Additional entries must be made for device codes that are to be treated as either TAPE1 or TAPE2. The format of the statement that adds entries to SASSUTBL is:

```

10  16
|   |
DC  CL8'xxxxxxxx',XL4'yyyyyyyy'
```

Where:

CL8'xxxxxxxx'

Indicates a required generic unit name which may be up to 8 characters. If CL4'DASD' is used instead of XL4'yyyyyyyy', then this is a volume serial number for data sets that are to be classified as DASD.

XL4'yyyyyyyy'

Indicates an 8-character, hexadecimal device code, as defined in the operating system device name table.

You may specify a reserved value of CL4'DASD' in place of the XL4'yyyyyyyy' to indicate that this entry refers to a volume serial number or unit name that is to be classified as DASD instead of tape.

Note: CA-7 always treats the ddnames of JOBLIB, STEPLIB, JOBCAT, and STEPCAT as DASD.

Examples: For example, assume four 2400s and four 3400s are to be TAPE1 and one 3420 is to be TAPE2. The entries may appear as follows, depending on the SYSGENed device codes:

```

10  16
|   |
DC  CL8'TAPE1',X'30808001' 2400
DC  CL8'TAPE1',X'30C08001' 2400
DC  CL8'TAPE1',X'34008001' 2400
DC  CL8'TAPE1',X'34208001' 2400
DC  CL8'TAPE1',X'30808003' 3400
DC  CL8'TAPE1',X'30C08003' 3400
DC  CL8'TAPE1',X'34008003' 3400
DC  CL8'TAPE1',X'34208003' 3400
DC  CL8'TAPE2',X'32108003' 3420
DC  X'FFFF'
```

Note: There must be an X'FFFF' terminator as the last entry.

When the hex device code or the IBM SYSGENed esoteric value is not known, a user-defined value can be specified as the hex device code for that unit. The third byte of the value should be a hex 80 to indicate a tape device. The CA-7 LOC command may be used to display the device codes for any currently cataloged data sets.

This same user-defined hex entry must then be placed in SASSUTBL with a unit name of TAPE1 or TAPE2. For example:

- UNIT=CART is coded in the JCL, rather than UNIT=3480
- The esoteric SYSGENed value for CART is not known
- CART is to be treated as a TAPE2 device. Then,

```
DC  CL8'CART',X'77778077'
DC  CL8'TAPE2',X'77778077'
```

In this example, any generic unit in SASSUTBL coded with a device code of X'77778077' is considered a TAPE2 device by CA-7 during the load process. The entry for TAPE2 cannot precede the generic unit (CART in this example).

In place of using TAPE1 or TAPE2, the value TAPE99 may be specified to cause CA-7 to treat a tape data set as being DASD and not count it as either TAPE1 or TAPE2. The following cause data sets using UNIT=CART to be treated as DASD files instead of counting as a tape file:

```
DC  CL8'CART',X'77778077'
DC  CL8'TAPE99',X'77778077'
```

At some sites, disk data sets may be archived to tape. The data sets should be classified as DASD, but the hex device code indicates TAPE. An entry can be specified in SASSUTBL to cause any data sets using a specified volume serial number to be classified as DASD instead of TAPE. Also, you can code entries to cause specific esoteric unit names to be classified as DASD. The SASSUTBL entries using the DASD designation should be coded before any other entries.

```
DC  CL8'ARCHIV ',CL4'DASD'  IF VOLSER=ARCHIV, ASSUME DASD
DC  CL8'MIGRAT ',CL4'DASD'  IF VOLSER=MIGRAT, ASSUME DASD
DC  CL8'SYS348XR',CL4'DASD' IF UNIT IS SYS348XR, ASSUME DASD
```

3.7.6 WLB Macros

The generation of workload balancing parameter definitions is accomplished through the assembly of the following special macros:

Macro	Description	On
WLBPDEF	Module name, required for format	3.7.7.1, “WLBPDEF” on page 3-57
TAPE1	Class of tape drives	3.7.7.2, “TAPE1” on page 3-58
TAPE2	Class of tape drives	3.7.7.3, “TAPE2” on page 3-60
CPU	CPU use	3.7.7.4, “CPU” on page 3-62
INTR	Initiator characteristics	3.7.7.5, “INTR” on page 3-63
CLBARR	Job class barriers	3.7.7.6, “CLBARR” on page 3-65
STARTIME	Scheduled start time for jobs	3.7.7.7, “STARTIME” on page 3-66
WLBPEND	Required for format	3.7.7.8, “WLBPEND” on page 3-67

The user must identify the different virtual configurations and processing objectives required in terms of macro parameter values. An assembly and link-edit generates a definition and stores it as a nonexecutable load member in the CA-7 load module library. (The link-edit should not use a PARM of RENT or REUS.) A permanent change requires macros to be recoded and assembled before an online session of CA-7 is started. Once the online session is started, temporary changes can be made with the XWLB and /WLB commands. These changes, however, are only temporary and do not affect the definition in the load library. They are erased by the next UCC7Rxxx definition loaded.

3.7.7 Coding WLB Macros

The WLBPDEF macro is required and must be coded first. The WLBPEND macro is also required and must be coded last. The other macros are optional and can be coded in any order. Coding rules for macros are as follows:

- The macro name must begin between columns 2 and 16.
- There must be at least one blank between the macro name and the first parameter, if coded.
- Continuation of any macro statement is indicated by a nonblank character in column 72. Macro parameters and values may appear through column 71 with no blanks. All continuation statements must begin in column 16. Keyword values may be broken for continuation at a comma. (This is a standard assembler continuation.)

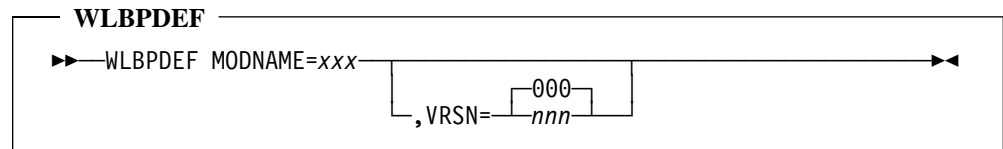
The detailed descriptions of these macros follow.

For assistance in creating the WLB macros, see the workload balancing objectives definition that is created when the workload balancing questionnaire is completed. The WLB questionnaire and a sample WLB Objectives Definition are presented in 3.7.4, “Workload Balancing Questionnaire” on page 3-45.

3.7.7.1 WLBPDEF

The macro provides a name and version number to a definition. The WLBPDEF macro is required and must be coded first; however, only code one macro per module.

Syntax



Where:

MODNAME

Specifies the last 3 characters of the name of a definition. The value must be alphanumeric. The name of the definition, then, becomes UCC7Rxxx. MODNAME=DFL should not be used for a user-defined objectives definition since this indicates the default and the objectives are ignored when selecting a job for submission.

VRSN

Specifies the optional version number, if any. The value must be numeric between 000 and 999. The default is 000. This field is for information only.

Examples:

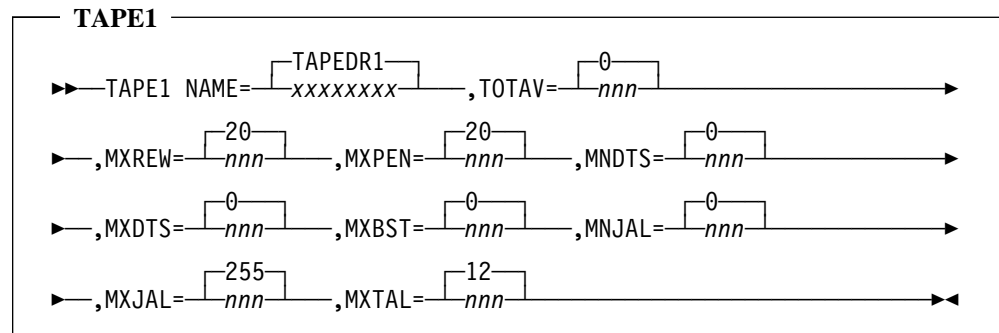
```
WLBPDF MODNAME=SAT,VRSN=001
```

The name of the generated definition is UCC7RSAT and the version number is 001.

3.7.7.2 TAPE1

TAPE1 provides information about tape drives of TYPE1. The TAPE1 macro is optional and can be coded anywhere between the WLBPDF and WLBPEND macros. Only code one macro per module.

Syntax



Where:

NAME

Provides a name for TYPE1 which may be up to 8 characters in length. For information only. Default is TAPEDR1.

TOTAV

Indicates the total number of TYPE1 tape drives available for scheduling. Must be numeric, between 0 and 255, and not higher than the value specified by MXTAL. Default is 0.

MXREW

Indicates the maximum reward to be given to a job using TYPE1 tape drives. Must be numeric and between 0 and 255. Default is 20.

MXPEN

Indicates the maximum penalty to be given to a job using TYPE1 tape drives. Must be numeric and between 0 and 255. Default is 20.

MNDTS

Indicates the minimum number of tape drives considered difficult to schedule. Must be numeric and between 0 and 255, and must not be higher than the value specified by MXDTS. Default is 0.

MXDTS

Indicates the maximum number of tape drives that are considered difficult to schedule. The value must be numeric, between 0 and 255, and not less than the value specified by MNDTS. Default is 0.

MXBST

Indicates the maximum boost to be added to the reward given to a job for difficult to schedule numbers for TYPE1 tape drives. Must be numeric and between 0 and 255. Default is 0.

MNJAL

Indicates the minimum number of TYPE1 tape drives allowed to be scheduled per job. Must be numeric, between 0 and 255, and not higher than the value specified by MXJAL. Default is 0.

MXJAL

Indicates the maximum number of TYPE1 tape drives allowed to be scheduled per job. Must be numeric, between 0 and 255, and not less than the value specified by MNJAL. Default is 255.

MXTAL

Indicates the total number of TYPE1 drives that can be scheduled. Must be numeric, between 0 and 255, and not less than the value specified by TOTAV. Default is 12.

Examples:

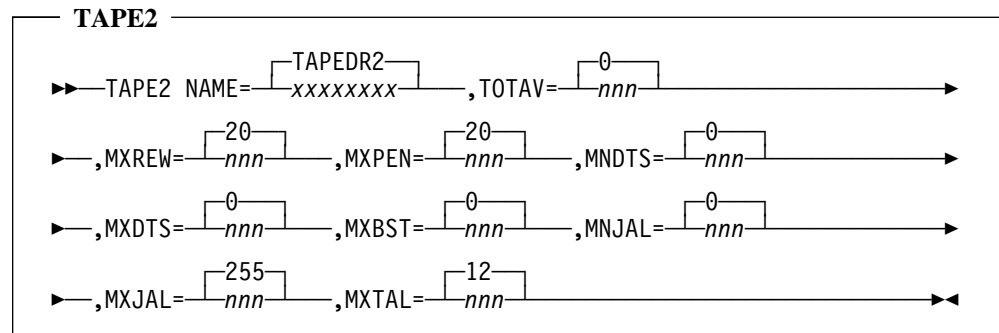
TAPE1	TOTAV=20,MXTAL=25,MXREW=20,MXPEN=15, MXJAL=10,MNJAL=0
-------	--

This example indicates that there are 20 TYPE1 tape drives under the name TAPEDR1 available for scheduling, although CA-7 can schedule up to 25 tape drives at any one time. If a job uses more than 10 tape drives, do not schedule that job even if the priority is highest. Under the most favorable conditions, give a maximum reward of 20 points to a job. Under the most unfavorable conditions, subtract 15 points from the priority of a job.

3.7.7.3 TAPE2

TAPE2 provides information about TYPE2 tape drives. The TAPE2 macro is optional and can be coded anywhere between the WLBPDEF and WLBPEND macros; however, only code one macro per module.

Syntax



Where:

NAME

Provides a name for TYPE2 tape drives which may be up to 8 characters in length. For information only. Default is TAPEDR2.

TOTAV

Indicates the total number of TYPE2 tape drives available for scheduling. Must be numeric, between 0 and 255, and not higher than the value specified by MXTAL. Default is 0.

MXREW

Indicates the maximum reward to be given to a job using TYPE2 tape drives. Must be numeric and between 0 and 255. Default is 20.

MXPEN

Indicates the maximum penalty to be given to a job using TYPE2 tape drives. Must be numeric and between 0 and 255. Default is 20.

MNDTS

Indicates the minimum number of tape drives considered difficult to schedule. Must be numeric, between 0 and 255, and not higher than the value specified by MXDTS. Default is 0.

MXDTS

Indicates the maximum number of tape drives considered difficult to schedule. The value must be numeric, between 0 and 255, and not less than the value specified by MNDTS. Default is 0.

MXBST

Indicates the maximum boost to be added to the reward given to a job using difficult to schedule numbers for TYPE2 tape drives. Must be numeric and between 0 and 255. Default is 0.

MNJAL

Indicates the minimum number of TYPE2 tape drives allowed to be scheduled per job. Must be numeric, between 0 and 255, and not higher than the value specified by MXJAL. Default is 0.

MXJAL

Indicates the maximum number of TYPE2 tape drives allowed to be scheduled per job. Must be numeric, between 0 and 255, and not less than the value specified by MNJAL. Default is 255.

MXTAL

Indicates the total number of TYPE2 drives that can be scheduled. Must be numeric, between 0 and 255, and not less than the value specified by TOTAV. Default is 12.

Examples:

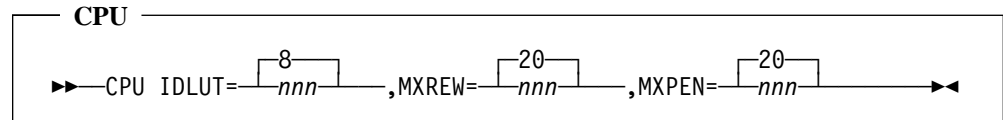
```
TAPE2  NAME=1600BPI,TOTAV=3,MXTAL=3,MNJAL=0
```

This example indicates that the name of the group of tape drives is 1600BPI for information and report headings only. There are only three tape drives available for scheduling. CA-7 cannot schedule more than three tape drives at any time. No job is to be held back because of minimum/maximum tape drives. Maximum reward and penalty are both 20 (default value).

3.7.7.4 CPU

The CPU macro provides information about CPU use. It is optional and can be coded anywhere between the WLBPDF and WLBPEND macros; however, only code one macro per module.

Syntax



Where:

IDLUT

Indicates the value of an ideal CPU use per job expressed as a percentage. Must be numeric and between 0 and 100. The product of the values specified by this keyword and by TOTAV in the INITR macro (following) must not be greater than 100. Default is 8.

MXREW

Indicates the maximum reward to be given to a job. Must be numeric and between 0 and 255. Default is 20.

MXPEN

Indicates the maximum penalty to be given to a job. Must be numeric and between 0 and 255. Default is 20.

Examples:

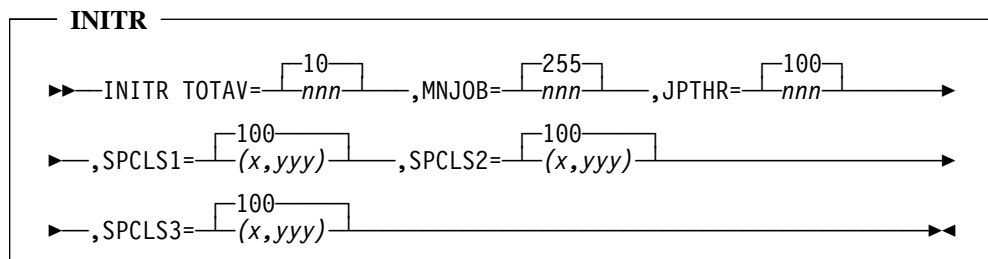
```
CPU  MXREW=15,MXPEN=10,IDLUT=10
```

This example indicates that CA-7 is to submit jobs in such a way that the percent CPU use per job is as close to 10 percent as possible. The job that would bring the CPU use per job closest to 10 percent gets the highest reward of 15 points. The job that would take the CPU use per job farthest from 10 percent (on either side) is penalized a maximum of 10 points.

3.7.7.5 INTR

INTR provides information about initiators. The INTR macro is optional and can be coded anywhere between the WLBDEF and WLBEND macros; however, only code one macro per module.

Syntax



Where:

TOTAV

Indicates the total initiators available for CA-7. Must be numeric and between 0 and 255. A value of 255 indicates there is not a limit. The product of the values specified by this keyword and by IDLUT in the CPU macro must not be greater than 100. Default is 10. A value of 255 indicates that there is not a limit.

MNJOB

Indicates the minimum number of jobs submitted concurrently by CA-7. If the current number of jobs submitted to the CPU falls below this value, jobs are submitted from the ready queue on a first in, first out basis within resource limits but regardless of priority calculations, until the total number of submitted jobs equals the MNJOB=value. Must be numeric and between 0 and 255. Default is 255.

JPTH

Indicates the default threshold priority. This value applies to all classes of jobs except any class specified in SPCLS1, SPCLS2, or SPCLS3. The threshold priority is the minimum calculated priority any job must have to be submitted. Even if a job has the highest priority of those ready to be submitted, if it is below the threshold for its class, it is not selected as the best job to be run at that time. Therefore, another class of jobs with a lower priority threshold is given preference for submission. Must be numeric and between 0 and 255. Default is 100.

SPCLS1

Indicates the first special class and threshold priority.

x

Indicates the job class. Must be a single alphanumeric character (A through Z or 0 through 9). No default.

yyy

Indicates the threshold priority associated with that class. Must be numeric and between 0 and 255. Default is 100.

Note: Jobs entering the request queue through a RUN command are assigned job class 9. LOAD jobs are assigned job class 8.

SPCLS2

Indicates the second special class and threshold priority.

x

Indicates the job class. Must be a single alphanumeric character (A through Z or 0 through 9). No default.

yyy

Indicates the threshold priority associated with that class. Must be numeric and between 0 and 255. Default is 100.

Note: Jobs entering the request queue through a RUN command are assigned job class 9. LOAD jobs are assigned job class 8.

SPCLS3

Indicates the third special class and threshold priority.

x

Indicates the job class. Must be a single alphanumeric character (A through Z or 0 through 9). No default.

yyy

Indicates the threshold priority associated with that class. Must be numeric and between 0 and 255. Default is 100.

Note: Jobs entering the request queue through a RUN command are assigned job class 9. LOAD jobs are assigned job class 8.

Examples:

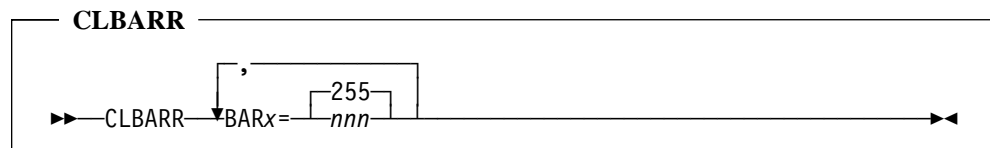
```
INITR  TOTAV=10,MNJOB=2,JPTH=200,SPCLS1=(B,50),
        SPCLS2=(0,100)
```

This example indicates that CA-7 can have up to 10 jobs running at a time. If the number of jobs running on a CPU under CA-7 control is less than two, CA-7 does not check the threshold priority until two jobs are running on the CPU. CA-7, of course, checks for any resource constraints (no initiators available, no tape drives available, and so forth). When at least two jobs are running, CA-7 does not submit any job from classes other than B and zero, unless the job priority (after all rewards and penalties are calculated) is 200 or more, and there are no resource constraints. If the job is from class B, the job priority must be 50 or above; if from class zero, 100 or above.

3.7.7.6 CLBARR

The CLBARR macro defines class barriers for CA-7 job classes. Each class barrier establishes the maximum number of jobs that can be submitted concurrently in the associated job class. The CLBARR macro is optional and can be coded anywhere between the WLPDEF and WLPEND macros; however, only code one macro per module.

Syntax



Where:

BAR_x

Barrier value for class x (see the **SPCLS1** keyword for 3.7.7.5, “**INTR**” on page 3-63) may have a value of A through Z or 0 through 9. The parameter value must be numeric and between 0 and 255. A value of 255 indicates there is not a limit. Multiple barrier values may be specified. Default is 255 for each.

Examples:

CLBARR BARA=5,BAR2=1,BARZ=0

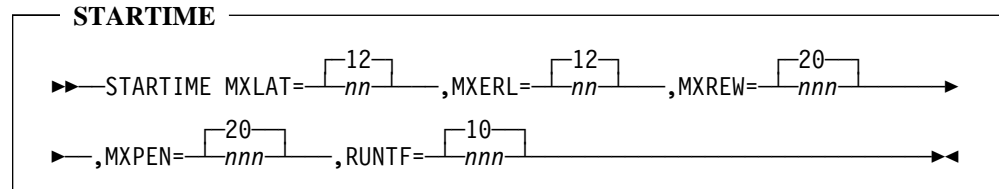
In this example, CA-7 can have only five jobs running under class A and only one job running under class 2. No job under class Z is ever to be submitted. In other classes, CA-7 can have up to 255 jobs running.

Note: Jobs entering the request queue through a RUN command are assigned job class 9. LOAD jobs are assigned job class 8.

3.7.7.7 STARTIME

The STARTIME macro provides information on the relative importance of the scheduled start time for a job. This macro is optional and can be coded anywhere between the WLBPDF and WLBPEND macros; however, only code one macro per module.

Syntax



Where:

MXLAT

Indicates maximum hours a job is considered late for the reward calculation. If a job is late by more hours than this value, it has the same effect as if it is late by maximum hours. That is, it receives the maximum reward (MXREW) for being late. Must be numeric and between 0 and 12. Default is 12 hours.

MXERL

Indicates maximum hours a job is considered early for the penalty calculation. If a job is early by more hours than this value, it has the same effect as if it is early by maximum hours. That is, it receives the maximum penalty (MXPEN) for being early. Must be numeric and between 0 and 12. Default is 12 hours.

MXREW

Indicates the maximum reward to be given a job which is late, based on the maximum number of hours. Must be numeric and between 0 and 255. Default is 20.

MXPEN

Indicates the maximum penalty to be given a job which is early by the maximum number of hours. Must be numeric and between 0 and 255. Default is 20.

RUNTF

The run time factor defines a tolerance to be allowed in determining whether a job is late or early. The factor is expressed as a percentage of the estimated job elapsed time. It must be numeric and range from 0 to 100. The default value is 10 (for 10 percent). A value of 0 indicates no tolerance is to be allowed.

Examples:

```
STARTIME    MXREW=30,MXPEN=10,MXLAT=6,MXERL=10
```

This example indicates that if a job is late by six or more hours, the reward is 30 points. For a job which is late between zero and six hours, the reward is proportionate based on the run time factor (RUNTF). For example, if late by three hours, the reward is 15. If late by one hour, reward is five. If a job is early by 10 or more hours the penalty is 10 points. If a job is five hours early the penalty is five points. If early by 15 hours the penalty is 10 points. By default, the run time factor is 10. This means a 10-hour running job is not considered late (and hence rewarded) unless it is late by one or more hours. A 5-hour running job is not considered early (and hence penalized) unless it is early by 30 minutes or more.

3.7.7.8 WLBPEND

WLBPEND indicates the end of a workload balancing parameter definition. The WLBPEND macro is required and must be coded last; however, only code one macro per module.

Syntax

WLBPEND

▶▶—WLBPEND—▶▶

There are no other keywords.

3.7.8 Generating the Modules

CA-7 generates definitions based on the parameters defined to the WLB macros specifically provided for this purpose. These generated definitions become nonexecutable load modules and are stored in the CA-7 load module library. The names of these load modules must be names in the format UCC7Rxxx where xxx can be any alphanumeric characters. No other jobs should begin with these same 5 characters. UCC7RDFL is a reserved name for the default definition module which is included on the version tape.

The following displays an entire workload balancing parameter definition for sample module UCC7R010. (A PRINT NOGEN is suggested because the macro expansions can be quite lengthy.)

10	16	72
PRINT	NOGEN	
WLBDEF	MODNAME=010	
TAPE1	NAME=TYPE1,	X
	TOTAV=25,MAXTAL=30,	X
	MXREW=20,MPEN=20,	X
	MNJAL=0,MXJAL=6,	X
	MNDTS=4,MXDTS=6,MXBST=10	X
CPU	MXREW=20,MPEN=20,IDLUT=5	
INITR	TOTAV=12,MNJOB=3,JPTH=100	
CLBARR	BARA=10,BARB=2	
STARTIME	MXLAT=6,MXERL=6,RUNTF=10,	X
	MXREW=20,MPEN=20	
WLBEND		
END		

3.7.9 Scheduling WLB Modules

With every workload balancing parameter definition, a job with the same name as the load module must be added to the database. These job names must begin with UCC7R followed by three unique characters. When a certain WLB definition is to be in effect, the job may be demanded in.

WLB modules may be scheduled like any other job in CA-7. It may have a calendar based schedule, it may be triggered (by either job completion, network completion or data set creation) or it can be DEMANDED. When all requirements are satisfied, the WLB definition with the same name as the job name is loaded and the criteria defined in that load module is in effect. The UCC7R job immediately goes through normal job completion and is never submitted to JES.

At all times that CA-7 is active, some definition must be present. In the absence of another definition, the default definition, UCC7RDFL, provided with the version tape, is used.

Note: If using the default definition, UCC7RDFL, none of the workload balancing procedures is in effect.

Job names beginning with UCC7R are reserved for workload balancing. If a job beginning with UCC7R enters the request queue, CA-7 attempts to find a load module with that name to use as the new workload balancing module. If unable to find it, message SCRJ-60 is issued to the master station and the job remains in the request queue with a master count of 1 and a condition code of 0.

A SUBMIT time requirement can be established for the WLB module, and then the module is not in effect until that SUBMIT time is satisfied.

3.7.10 Workload Balancing Commands

Command	Use
/WLB	Use the /WLB command to activate, deactivate, or modify the workload balancing function of CA-7.
LWLB	Use the LWLB command to list currently active workload balancing and performance management processing objective information.
RESCHNG	Use the RESCHNG command in a CA-7 trailer step to free tape drives that are no longer needed.
XWLB	Use the XWLB command, included in the workload balancing facility, for making temporary changes in selection parameters.

See the *CA-7 Commands Guide* for more information on these commands.

3.8 Tracking External Tasks

CA-7 has the ability to track batch jobs and started tasks that are submitted outside of CA-7. Throughout the documentation of external tracking, the term task is used to refer to either a batch job or started task. This does not include TSO users. The tracking is selective based on entries in two tables that are coded to specify the task names which are to be tracked. This is only available for CPUs that share the communications data set. In an NCF environment, the external tracking is only available for NCF1 sites. Also, an NCF1 site can track only external jobs that execute at its own site, not at another NCF1 site.

To track external tasks, you must code two tables, SASSEXTL and SASSEXTT. SASSEXTL is used by the CA-7 SMF collection process to determine which external tasks are tracked. SASSEXTT is used by CA-7 to build the queue record for the external task. The SASSEXTL code is "activated" during the CAS9 (CAIRIM) process.

When an external task is tracked, CA-7 is monitoring task start, step terminations, and task termination. The data set information is not collected. An external task is represented by a record in the CA-7 queues, dynamically added when CA-7 gets the task start SMF record. When the task successfully terminates, it is posted to the CA-7 run log and, optionally, prior-run queue and database.

External tasks do not need to be defined to the CA-7 database. However, if these tasks are defined to CA-7, they can be used for triggering and predecessor relationships, except for negative dependencies which cannot be used.

3.8.1 Usage Notes

Tracking external tasks can have an impact on CA-7 performance. This impact is because of the additional SMF records that are processed.

Because external task completions are kept in the run log, the trailer queue may fill more quickly than if only CA-7 tasks are tracked.

The active queue may need to be allocated larger to accommodate the external work that will be represented in the queue.

If an external task terminates unsuccessfully, it remains in the CA-7 request queue with a restart status. The task has to be restarted outside of CA-7, and then the CA-7 queue record has to be CANCELED. If the task is to be considered successfully completed, use the topline RESTART command to force complete the job.

External tasks are not included in workload balancing, even though they are in the active queue.

External tasks cannot be used for negative dependencies.

External task tracking is not available for the test copy of CA-7.

3.8.2 Defining External Tasks To Be Tracked

The module SASSEXTL defines a table of task names that SASSU84 uses in determining which external tasks to track. The CAIRIM initialization for CA-7 (procedure CAS9) loads the table. If SASSU84 finds a match in the SASSEXTL module, it collects all task initiation, step termination, and task termination SMF records for the task. To track external jobs, SMF type 30 support IS REQUIRED.

Note: NCF2 sites cannot track external work. Therefore, those sites should NOT have a SASSEXTL module. Also, NCF1 sites can track only the external work that executes at their own site.

After coding the module, you must assemble and link to the library that the CAS9 procedure (CAIRIM) uses to initialize CA-7. The CAS9 procedure must be run to load the SASSEXTL module.

Entries in SASSEXTL should be coded using the following format:

column	
10	16
DC	CL8'xxxxxxxx'

Where:

CL8'xxxxxxxx'

Indicates a required task name which may be from 1 to 8 alphanumeric characters.

This may also include ? to mask a character or * to terminate a generic specification.

See member UL23334 in SAMPJCL for a sample SASSEXTL module in SMP format.

3.8.2.1 SASSEXTL Example

```

column
10      16
|       |
TITLE  'CA-7 EXTERNAL TASK FILTER TABLE'
SPACE  1
SASSEXTL START
DC      CL8'TESTJOB*'
DC      CL8'TEST???X'
DC      CL8'TESTSTC '
TABEND  DC      4X'FF'          END OF TABLE INDICATOR (REQUIRED)
END

```

Note: You must define each entry in the table as CL8 (padded with blanks if necessary), and place an x'FF' terminator as the last entry.

In the above example, the SASSEXTL table causes SASSU84 to collect SMF data for tasks with the criteria:

- First 7 characters of name are equal to TESTJOB
OR
- First 4 characters of name are equal to TEST and 8th character equal to X
OR
- Task name equal to TESTSTC

3.8.3 Defining the Model Queue Records

The module SASSEXTT defines a table of model queue records that are used for external tasks. If this load module does not exist, CA-7 does not post the CA-7 queues for external jobs, even if the SMF data is being collected.

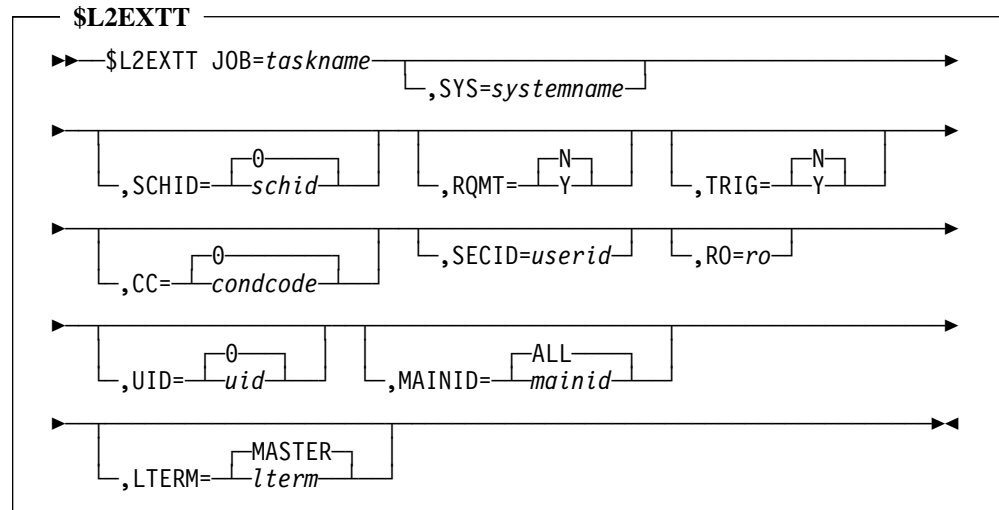
When the task start SMF record is received, CA-7 recognizes that the task was submitted externally and adds a record to the active queue for the task. If there is already a record in the active or ready queue for this task, an error message is issued (to the master station and also as a WTO), and the new queue record is not added.

When CA-7 adds a record to the active queue for an external task, it searches the SASSEXTT module for a model which specifies the information used for that task. CA-7 uses the FIRST entry in SASSEXTT that matches the task name (not necessarily the best match or most specific). If the SASSEXTT module does not contain a match, then CA-7 builds the queue record using all default values.

3.8.4 \$L2EXTT Macro

The \$L2EXTT macro is used in the SASSEXTT module to define the model queue records used when tracking external tasks.

Syntax



Where:

JOB

Indicates a required task name specification. This is used to match a model queue record with an external task name. The name may be specific, generic, or masked using a question mark. If using masking or generics, code the \$L2EXTT calls in the order of most specific to least specific. CA-7 uses the first entry in the table that matches an external task name as the model queue record.

Size/Type: 1 to 8 alphanumeric characters (may be terminated with an asterisk or specify a question mark to mask a character)

Required: Yes

Examples:

\$L2EXTT JOB=ABC*

All tasks whose first 3 characters are ABC

\$L2EXTT JOB=*

All tasks (should be coded last in the table if used)

\$L2EXTT JOB=A?C

Tasks whose first letter is A, third letter is C and have any valid character for the second letter (only includes 3-character task names)

\$L2EXTT JOB=A?C*

Same as previous entry but also includes tasks with more than 3-character names

SYS

Indicates the system name for the queue record.

Size/Type: 1 to 8 alphanumeric characters
 Default: Blanks
 Required: No

SCHID

Indicates the schedule ID value for the queue record.

Size/Type: 1 to 3 numeric characters from 0 to 255
 Default: 0
 Required: No

Note: If this value is set to zero, the schedule ID of the active queue records is set according to the global external job schedule ID parameter. This global parameter is set by the EXTSCHID= keyword on the Options statement in the CA-7 initialization file.

SECID

Indicates the security user ID associated with the task. This is available to be propagated to triggered jobs. This keyword has no effect unless CA-7 external security is used with SUBUID.

Size/Type: 1 to 8 alphanumeric characters
 Default: Blanks
 Required: No

RQMT

Specifies if this task can satisfy requirements for a job on CA-7. If Y is specified, the successful completion of this task posts any jobs that are currently in the CA-7 request queue waiting for the task. Also, if the task is defined to the CA-7 database, the successful completion updates the database information for the task.

Size/Type: N or Y
 Default: N
 Required: No

TRIG

Specifies if this task can be used to trigger work on CA-7. If Y is specified, RQMT=Y must also be specified.

Size/Type: N or Y
 Default: N
 Required: No

CC

Used with RO to define the job-level condition code used to determine if a task executes successfully or not.

Size/Type: 1 to 4 numeric characters from 0-4095
 Default: 0
 Required: No

RO

Indicates the relational operator of the condition code. The highest condition code that a task generates is tested by this pair of parameters (CC and RO). This test is for CA-7's internal use only. It simply tells CA-7 what action to take AFTER the task completes.

Size/Type: 2 alpha characters: EQ, LT, GT, GE, LE, NE, or IG
 Required: No

UID

The CA-7 security identification.

Size/Type: 1 to 3 numeric characters from 0 to 255
 Default: 0 (No UID security protection)
 Required: No

MAINID

This does not control where the task is executing. However it can be propagated to any triggered jobs if TRIG=Y is coded. Specify SYn or /SYn where n is the system number and / indicates not this system.

Size/Type: 1 to 4 alphanumeric characters
 Default: ALL
 Required: No

LTERM

Messages about this job are to be routed to this logical terminal name.

Size/Type: 1 to 8 alphanumeric characters
 Default: MASTER
 Required: No

3.8.5 Defining the SASSEXTT Module

The SASSEXTT module is comprised of 1 or more \$L2EXTT macro calls. After the module is coded, it must be assembled and linked.

If the size of the module exceeds 32K, you must include an APPLCTN statement in the CA-7 initialization file. The format is as follows:

APPLCTN,NAME=SASSEXTT,ATTR=PERM

It must be placed in front of the APPLCTN statement for SASSPROG.

See member UL23335 in SAMPJCL for a sample SASSEXTT module in SMP format.

3.8.5.1 SASSEXTT Example

```

column
10      16                                72
|        |                                |
|        |                                |
|        |                                |
SASSEXTT TITLE 'EXTERNAL JOBS MODEL QUEUE HEADER RECORDS'
START
PRINT ON
$L2EXTT JOB=TESTTEXT1,SYS=TEST,SCHID=3,RQMT=Y,TRIG=Y,          X
        CC=0,RO=LT
$L2EXTT JOB=TEST???1,SYS=TEST,SCHID=20,MAINID=SY1,RQMT=Y,      X
        CC=0,RO=LT
$L2EXTT JOB=TEST*,SCHID=5,CC=0,RO=LT
$L2EXTT JOB=*,SYS=TEST
TABEND DC 2X'FF'          END-OF-TABLE MARKER  **REQUIRED**
END
```

Note: There must be an x'FF' terminator as the last entry.

3.9 Tracking External Data Sets

CA-7 has the ability to track SMF data for data sets that are accessed by tasks not submitted by CA-7. The tracking process examines SMF type 15 records and compares the job name and data set name information with a table, SASSXDSN, which is created at each site.

SASSXDSN is used by the CA-7 SMF collection process to determine which external tasks are tracked. The tracking is done by capturing the SMF type 15 (x'0F') record which is generated when a data set is closed after being opened for output. Data set records can be captured for batch jobs as well as started tasks and TSO users. When a record is captured, CA-7 attempts to associate it with a job in the active queue. If the job which created the SMF record does not exist, CA-7 processes the data set as a POST AT CREATE TIME data set and updates the database with the date and time. If the creating job is in the active queue, the data set is stored with that job, and the database is updated when the job completes.

External data sets DO need to be defined to the CA-7 database to be used as requirements for jobs or for data set triggers.

3.9.1.1 Usage Notes

Tracking external data sets can have an impact on CA-7 performance. This impact is because of the additional SMF records that are processed.

If the external data set is associated with an external task that is in the active queue, then the model queue record for the job must have RQMT=Y specified in the SASSEXTT module. If this is not specified, the data set information is not posted to the database.

External data set tracking is not available for the test copy of CA-7.

External data set tracking is available only for CPUs that share the communications data set. In an NCF environment, this tracking is available only for the local NCF1 site.

External data sets must be physical sequential, permanent files (not temporary).

For a system (CPU/machine) that has a period of time where the SMF activity related to CA-7 submitted jobs is low, the external data set postings to CA-7 can be delayed. The length of the delay depends upon CA-7 SMF activity through ICOM. This includes (but is not limited to) when a CA-7 submitted job starts (or ends or has a step term) or any job that goes through a JES purge on the system where the data set was created.

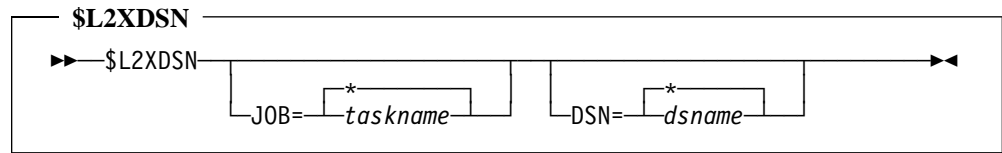
3.9.2 Defining External Data Sets To Be Tracked

The SASSXDSN module defines the criteria that SASSU83 uses to determine which external data sets are tracked. The table is loaded by the CAIRIM initialization for CA-7 (procedure CAS9). If SASSU83 finds a match in the SASSXDSN module, the SMF type 15 record is collected for the data set.

Note: External data sets cannot be tracked at NCF2 sites. Therefore, there should NOT be a SASSXDSN module at those sites. Also, NCF1 sites can track only external data sets that are created at their own site.

3.9.3 \$L2XDSN Macro

The `$L2XDSN` macro is used in the `SASSXDSN` module to define the filtering criteria for external data sets.



Where:

JOB

Indicates a task name specification. This is used to match the task name in the SMF record. The name may be specific, generic, or masked using a question mark.

Default: *

Required: No

Size/Type: 1 to 8 alphanumeric characters (may be terminated with an asterisk or specify a question mark to mask a character)

DSN

Indicates the data set name. This is used to match the data set name in the SMF record. The name may be specific, generic, or masked using a question mark.

Default: *

Required: No

Size/Type: 1 to 44 alphanumeric characters with a ? (question mark) used to mask a character, or use multiple question marks to mask multiple characters. You can use an * (asterisk) to terminate the matching. Thus, only one * is allowed, and it must be last if used.

Examples:

```
$L2XDSN JOB=ABCDE
```

All data set SMF 15 records for job ABCDE.

```
$L2XDSN DSN= A.B.C
```

All data set SMF 15 records for A.B.C, regardless of the task that generates the records.

```
$L2XDSN JOB=STC1,DSN=TRANSMIT.*
```

Data set SMF 15 records for data sets that have a high-level node of TRANSMIT and are generated by task STC1.

```
$L2XDSN DSN=PROD.D??? .DATA
```

Data set SMF 15 records for data sets that match this name, with any character in the positions of the ???, regardless of the task that generates the records.

3.9.4 Defining the SASSXDSN Module

The SASSXDSN module is comprised of 1 or more \$L2XDSN macro calls. After the module is coded, it must be assembled and linked.

See member UL23337 in SAMPJCL for a sample SMP USERMOD to create the SASSXDSN module.

3.9.4.1 SASSXDSN Example

```

column
10      16
|       |
|       |
TITLE 'EXTERNAL DATA SET TRACKING CRITERIA'
SASSXDSN START
PRINT ON
$L2XDSN JOB=ABCDE
$L2XDSN DSN=A.B.C
$L2XDSN JOB=STC1,DSN=TRANSMIT.*
$L2XDSN DSN=PROD.D??? .DATA
END
```

Note: If you are tracking external jobs/tasks using SASSEXTL and SASSEXTT, and also tracking data sets in those jobs/tasks, you must specify RQMT=Y and TRIG=Y in the SASSEXTT definitions for those jobs/tasks.

3.10 UNIX System Services Interface

The UNIX System Services Interface provides a means to communicate with CA-7 from the UNIX environment on OS/390. There are two CA-7 modules that must reside on the UNIX file system to ensure proper execution of the interface. The modules are CA7OECOM and CA7OESTB. This topic outlines the steps necessary to install the interface.

Copy the CA7OECOM module to the UNIX file system (HFS). The module resides in the CA-7 installation MACLIB (CAIMAC) in object format and must be copied to the UNIX file system (HFS) using the IBM Open Edition OPUT command. See the following example :

```
OPUT 'CA7.INSTALL.CAIMAC(CA7OECOM)' '/users/bin/ca7oecom.o' BINARY
```

Where:

OPUT

The command used to copy objects to the UNIX file system.

'CA7.INSTALL.CAIMAC(CA7OECOM)'

This is the OS/390 data set where the CA7OECOM member resides.

'/users/bin/ca7oecom.o'

This is the path where the object module will be copied. The ca7oecom.o specification, in lowercase, identifies the module name when it is copied to the UNIX file system. **The ".o" extension on the file name is required.**

BINARY

This option indicates that the object contains binary data and should be copied as such. **This option is required.**

Once the CA7OECOM object is copied to the UNIX file system, it must be link edited into executable format. The CA-7 installation SAMPJCL file contains a member USSLINK which can be used to perform the link-edit. The USSLINK JCL must be modified to your installation standards prior to execution. The JCL executes the IBM BPXBATCH program, which is the batch interface to UNIX System Services. The PARM passed to the program contains a link-edit command that must be modified to include the path (location) of the ca7oecom.o object you copied in the previous step.

```
PARM='sh c89 -v -o /xxxxxx/xxxx/ca7oecom /xxxxxx/xxxx/ca7oecom.o'
```

The /xxxxxx/xxxx/ca7oecom indicates the path where the executable ca7oecom will reside after the link-edit and what the executable name will be (ca7oecom). The /xxxxxx/xxxx/ca7oecom.o indicates the path or location to the ca7oecom.o object copied to the UNIX file system in the previous step.

The next step is to copy module CA7OESTB to a PDS/E library and then copy it from the PDS/E to the UNIX file system. This procedure creates an object of the executable and prepares it for execution on the UNIX file system. The CA-7 SAMPJCL library member USSCOPY contains JCL that can be used to allocate a PDS/E and then copy the module CA7OESTB to the PDS/E. Once the module resides in a PDS/E you can copy the module to the UNIX file system using the Open Edition OPUT command. See the following example:

```
OPUT 'CAI.PDSE.LOAD(CA7OESTB)' '/users/bin/ca7oestb' BINARY
```

3.10.1 File Permissions

In a UNIX environment, file security is determined by a set of permissions associated with a file. The CA-7 Interface modules must be marked as "executable" to allow users to invoke the interface. Below is an example output from a "ls -l" command (List) which show the file permissions associated with the two interface modules. You can use the UNIX "chmod" command to change file permissions.

```
-rwx---x--x  1 U01USER  AWORKSG    65600   Jul  6 1998  ca7oecom
-rwx---x--x  1 U01USER  AWORKSG    4096    Feb  3 16:16  ca7oestb
```

Chapter 4. Installation and Maintenance Procedures

This chapter describes the general installation procedures and postinstallation testing.

The installation and maintenance of CA-7 is performed through SMP/E. See the *CA-7 Getting Started* guide for detailed installation steps.

The final step in any installation process is to test the newly installed system. You should have been directed to this section by the *CA-7 Getting Started* guide. Ensure that all preceding installation steps have been completed successfully prior to performing the installation verification.

4.1 Installation Verification

The job names referenced in this section begin with CA07 unless a different 4-character prefix was assigned in the CA-7 SYSGEN macros. The last 4 characters are Nxxx where xxx is a unique number.

The installation verification process involves several steps:

1. Execute CA-7 online (job CA07N240).
2. Execute CA-7 ICOM (job CA07N500).
3. "Demand" installation verification job (job CA07SVCT, U7TESTSV program).
4. "Demand" the test network (jobs CA07XX01-CA07XX10).
5. Force CA-7 log dump.
6. Check CA-7 report writing using CA-Earl.

4.1.1 Step 1: Execute CA-7 Online

Job CA07N240 in the CA-7 JCLLIB is an online execution of CA-7. The first time CA-7 online is executed, TYPE=FORM must be specified on the EXEC statement. Also, you should use TIME=1440 on the JOB statement since CA-7 online should remain up during most of the day.

Submit job CA07N240 and pay careful attention to the startup messages on the system console. When the message CA-7.993 - CA-7 INITIALIZATION COMPLETE appears, CA-7 online is up.

If the CA-7.V001 message appears indicating a CONNECTION FAILURE, it is probably because the VTAM terminal ID specified in the SYSGEN macros was defined or defaulted incorrectly. This can be corrected by updating the first TERM statement (VTAMID=) in the CA-7 online initialization file (member ONLINE in the CA-7 JCLLIB) with an actual VTAM terminal ID. You can still log on to CA-7 even though this message appears, but it should be corrected before considering production work.

4.1.2 Step 2: Execute CA-7 ICOM

Job CA07N500 in the CA-7 JCLLIB is an execution of the CA-7 Independent Communications facility (ICOM). This job should also use TIME=1440 on the JOB statement since ICOM should remain up at all times. When the message CA-7.574 ENTER ICOM REQUEST appears, ICOM is active.

4.1.3 Step 3: 'Demand' Installation Verification Job

While both CA-7 and ICOM are running on the same CPU, the installation verification job should be run (job CA07SVCT, U7TESTSV program) to verify that CA-7 has been installed correctly. This job must be submitted by CA-7.

Log on to CA-7 through the application ID defined in the preinstallation instructions (default = CA7). When the CA-7 logon screen appears, enter MASTER for the USERID and no password. This logs you on to CA-7 using the installation-supplied default security.

Issue the following command to submit the installation verification job:

```
DEMAND,JOB=CA07SVCT      (job name prefix may have been changed)
```

Abend Codes: Normal termination of U7TESTSV is a U0099 abend code. Otherwise a U0098 abend code is generated. If a U0098 abend occurs, check the job's JES log for any messages that were issued. These messages indicate the reason for the U0098 abend. In any case, if the LQ command shows the job STATUS as abended, then tracking is working correctly, and the next postinstallation step should be done.

Messages: All messages generated by the program are issued through WTOs. See the *CA-7 Message Guide* for further information on U7TESTSV messages. Retain the abend dumps from execution on each CPU.

Reports: In addition to the WTOs, a report is produced if a SYSPRINT DD statement is included in the U7TESTSV JCL. This report summarizes important ICMDSECT information relating to the installation. The SMF selectivity tables are read to report on the SMF record types needed for proper CA-7 execution and the entry point addresses for the SMF interface exits are listed. A sample U7TESTSV report follows.

yyddd	10:24:24:74	ICMDSECT OPTIONS	
	STEPLIB	LPA/CSA	
TYPE 30 SUPPORT	Y	Y	
NCF	N	N	
USERID/RDR FIELD	7	7	
SVC NO	167	167	
JOB ID(HEX)	31	31	
IND BYTE(HEX)	EE	EE	
TEST SVC NO	---	---	
SUBPOOL NO	241	241	

SMF INFORMATION			
ADDRESS RETURNED FROM LOAD FOR UJV EXIT IS 821C9910.			
ADDRESS RETURNED FROM LOAD FOR U83 EXIT IS 821C9508.			
ADDRESS RETURNED FROM LOAD FOR U84 EXIT IS 821C90E8.			

Figure 4-1. Sample U7TESTSV Report

If U7TESTSV indicates that the installation is correct but jobs are still not being tracked by CA-7, validate the following:

- The initialization file SVCNO statement indicates SASSVC=YES.
- All queue data sets are at least 3 contiguous tracks.
- The SMF record types needed by CA-7 are being generated and the SMF interface exits are getting control. Use the OS console command D SMF,O to verify. Also see the CAIRIMU procedure/job.
- If a job is submitted with a step name of CA7B\$\$4U, WTOs are issued from each of the CA-7 SMF exits when they receive control. The messages are CA-7.Uxx - SASSUxx ENTERED where xx is JV, 83, or 84.
- The control report produced by U7TESTSV indicates the correct record types and that the type 30 support is consistent in all SMF interface exits and ICMDSECT.
- All of ICOM's PARMs indicate that SMF collection is active. Also, all of ICOM's JCL points to the same communications data set as CA-7.
- The same STEPLIB (CA-7 loadlib) is used by both CA-7 and all ICOMs.

If you are scheduling for a multiple CPU environment, this test job should be run on each CPU to verify that tracking is functional for all machines. This can be done by issuing DEMANDH as above, then using QJCL,n where n is the job number returned by the command. QJCL puts you in EDIT mode for the job where you can insert ROUTE statements to cause the job to execute on the target CPU. After inserting the ROUTE statement, enter SS which saves your update, then issue RELEASE,JOB=n where n is the job number mentioned above. Be sure that CAIRIM has been set up on each CPU and that ICOM is active on each one before DEMANDING these jobs. Save the dumps from this job as run on each CPU.

If CA-7 NCF is installed, the U7TESTSV program always abends with a U0098 unless a character 7 is placed in column 69 of the job statement for the job that executes U7TESTSV. The modified job statement allows the U7TESTSV program to do checking and complete with a U0099 abend if the installation is correct. However, the job(s) that executes at the remote NCF site(s) does not track through the CA-7 queues when the job statement has a 7 in column 69 even if the completion is a U0099.

4.1.4 Step 4: 'Demand' the Test Network

Once you have checked job tracking as described above, you are ready to execute the test job network which exercises other scheduling options and allows you to watch CA-7 perform. This can be done by issuing the following commands:

```
DEMAND,JOB=CA07XX01
and
DEMAND,JOB=CA07XX08
```

These commands schedule two of the jobs defined by installation job CA07N220. These jobs in turn cause other jobs to be scheduled as noted in the following figure. To monitor the progress of the jobs as they execute, issue the LQ command every few seconds and notice how the display changes. After all the jobs have completed, you may issue the LRLOG command to see the history of their scheduling, and you may wish to issue the LPRRN command as well.

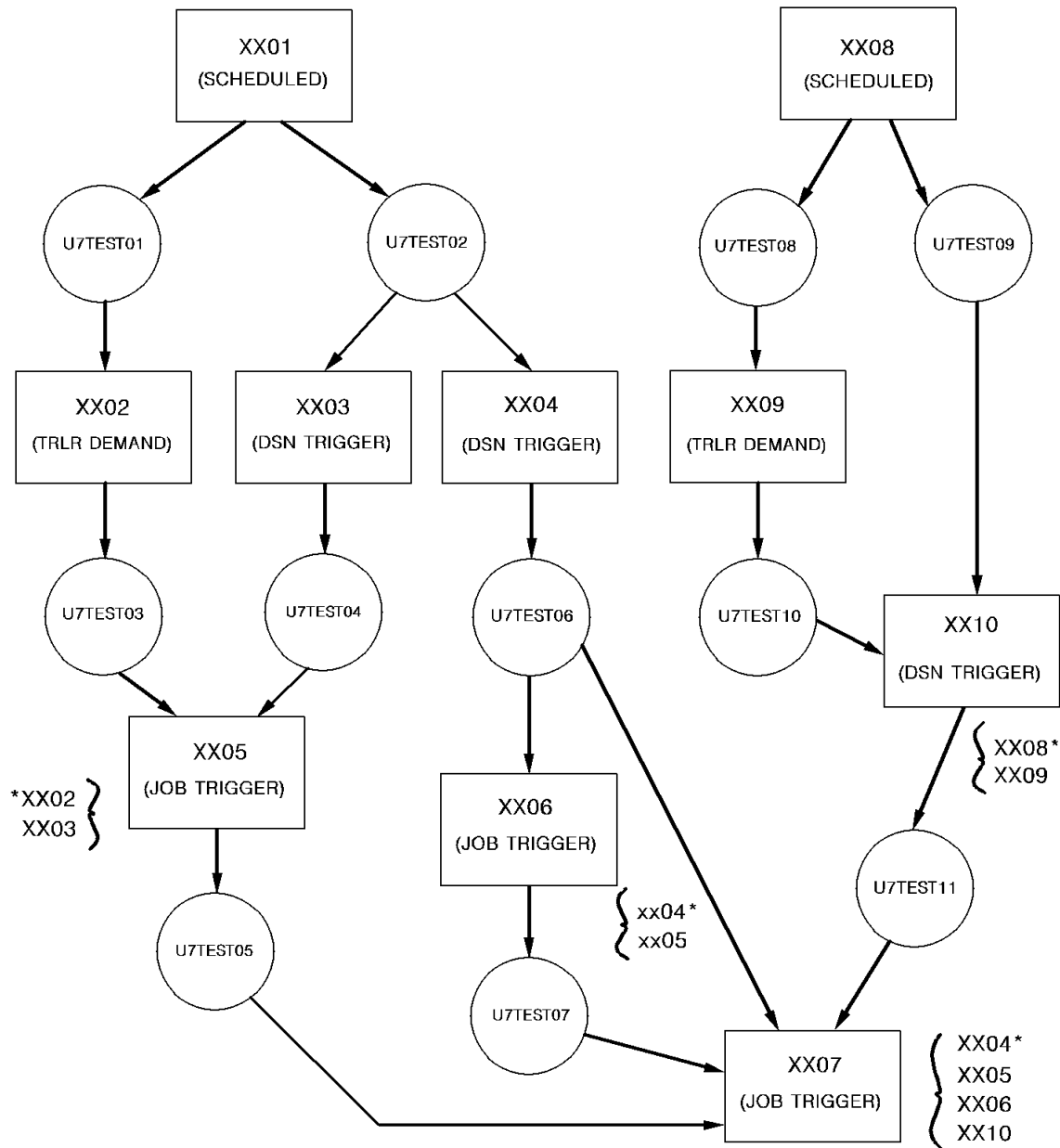


Figure 4-2. Test Network Execution

* Jobs in this group denote job dependencies.

4.1.5 Step 5: Force CA-7 Log Dump

Issue the CA-7 /SWAP command to force scheduling of a log dump job (CA07LOGP). This job dumps the primary log (LOGP) to the log history tape file and switch CA-7 processing to the secondary log file (LOGS).

After the log dump job completes successfully, go to the next verification step.

4.1.6 Step 6: Check CA-7 Report Writing Using CA-Earl

The CA-7 sample JCL library used in the installation process contains member L233EARL. This is a sample execution of the CA-7 report writing facility using the CA-Earl component distributed with Unicenter TNG Framework for OS/390.

Edit member L233EARL and provide a valid JOB statement for your system. Read the comment box and supply the procedure variables described. Submit job L233EARL and check the output. The job creates a CA-7 Job Scheduling/Completion Activity report detailing the processes generated by running the installation test job and the test job network as described in the preceding steps.

For detailed information on the CA-7 CA-Earl interface and the standard reports available, see the *CA-7 Reports Guide*.

4.2 Test Copy of CA-7

Usually only one CA-7 is executed at any one time within a single computing environment. The environment can consist of a single CPU or several CPUs. The limiting factor for the number of CPUs is the communications data set, which must be shared by all the CPUs. Running two copies of CA-7 can be helpful when upgrading to a new version (see note below), while applying maintenance or testing user exits.

Caution

There is only one copy of the CA-7 SMF interface exits and the SVC modules (other than the first load). Therefore, these modules cannot be tested using this method. This means that these modules must be installed for the production level of CA-7. The test copy may be a different version level, but the CA-7 system interface modules are at the production level.

4.2.1 Test Copy Implementation Steps

1. Allocate all new CA-7 data sets for the second copy of CA-7 (CAISRC, CAIMAC, CAILIB, and SAMPJCL do NOT have to be duplicated). The easiest way to do this is to run a second CA-7 SYSGEN to create a test version of JCLLIB.
 - a. In SAMPJCL, make a copy of L233GEN naming it L233TGEN. On the SYSPUNCH DD statement, change the name of the SAMPJCL member from STAGE1 to TSTAGE1. Make the following changes to the SYSGEN macros:

U7GEN macro

- Add/change the TESTSYS= keyword value to YES. For example:
TESTSYS=YES.

U7PARMS macro

- If you have a TARGET= keyword, do not change it. If you do not have one, code one using the DSN prefix of your production target libraries (CAILIB, CAISRC, and so forth).
- Add/change the NODE= data set name prefix to one different than your production prefix. For example: **NODE='CALCA7TEST'.**
- If you have a VSAM= keyword, change the data set name prefix to one different than your production prefix. For example:
VSAM='CALCA7TEST.VSAM'.

U7PNAMES macro

- If you wish to automatically generate a test copy set of JCL procedures, add/change the U7PNAMES macro with a different prefix. For example: **'U7PNAMES PREFIX=CA7T'**.

U7IFACE macro

- Add/change the VTAM= keyword to specify a different application ID for the test copy. For example: **'U7IFACE VTAM=(CA7T)'**.

U7VOL and U7SPACE macros

- If you wish to change the DASD packs or sizes of the test copy set you can use these macros to change those settings.

- Run the L233TGEN job to generate the TSTAGE1 SYSGEN deck. Check to make certain the data set name for JCLLIB in the first step is different than your production copy.
- Run the TSTAGE1 job (CA07N000) to create a test copy of the JCLLIB library.
- The following jobs need to be run from your TEST JCLLIB:

CA07N010 - Allocates test copy files
 CA07N030 - Allocate / Initialize test copy files
 CA07N720 - Copy help members to test HELP data set

- Copy the VTAM definition members from the test JCLLIB to your system VTAMLST library. Copy member CA7VTAM renaming it to CA7VTAMT. Copy member CA7ISPF renaming it CA7ISPFT. You need to vary these active before you can use the test copy of CA-7 (that is, **V NET,ID=CA7VTAMT,ACT**).
- Another SVC number is required to run the second copy. Identify an available user SVC number. As with the production SVC, the test SVC is dynamically added to the system when CAIRIM is run to initialize CA-7. However, you must ensure the test SVC number assigned is not currently being used.
- Set the test SVC number in the production copy of ICMDSCT. See member UL233IZ in the CA-7 SAMPJCL library for instructions to reapply the Usermod to set ICMDSCT options. The zap for the test SVC is at displacement 0E.

Normally the Test copy of CA-7 does NOT receive SMF purge record feedback. This means that certain JCL Errors on jobs it submits are not detected by the Test copy. You can force these purge records to be sent to the Test copy as well as the Production copy by setting the X'02' bit on at offset X'07' in the Production ICMDSCT.

Note: Enabling purge records for the Test copy will cause additional CSA and CPU overhead for the CA-7 system interfaces since two copies of ALL Purge records will be processed.

5. If your CA-7 load library (CAILIB) is different from the Unicenter TNG Framework for OS/390 load library used in the CAIRIM procedure (CAS9), you should copy member ICMDSECT from the CA-7 load library to the Unicenter TNG Framework for OS/390 load library replacing the existing version.

If you wish to test a new copy of the first load of the CA-7 SVC (module IGCS0XXX), copy (or link edit) the test module into the Unicenter TNG Framework for OS/390 CAILIB RENAMING IT TO IGCS0XX2.

6. After the previous steps have been performed, CA-7 needs to be reinitialized to bring in the new ICMDSECT and set up the test copy system interface. You can do this in one of two ways:
 - a. IPL and allow CAIRIM to initialize CA-7 as usual. The new test SVC number in module ICMDSECT is detected, and the test system interface is initialized (see details below).
 - or
 - b. Perform a CA-7 COLD reinitialization with CAIRIM. This causes a new production ICMDSECT to be brought into common storage. The new test SVC number is detected, and the test system interface is initialized (see details below). See 6.7, "CAIRIM Initialization Considerations" on page 6-38 for instructions on how to perform a COLD reinitialization of CA-7.

Note: Use extreme caution when performing a CA-7 COLD reinitialization.

7. When CAIRIM initializes a CA-7 test system the following steps are performed:
 - a. A subsystem SSCT entry is created for the CA-7 test copy (SSCT=UCT7).
 - b. A second copy of ICMDSECT is brought into common storage under the name ICMDSEC2. Appropriate changes are made to ICMDSEC2 to make it function as a test copy.
 - c. The test SVC module (IGCS0XX2) is loaded into common storage. If no IGCS0XX2 is found in CAILIB a second copy of the production SVC module (IGCS0XXX) is loaded. In either case, the test SVC module is loaded into common storage under the proper SVC module name (IGC00xxx). The module is modified to make it function as a test copy. The test SVC module is then engaged in the system under the test SVC number specified in the production ICMDSECT.
 - d. If all of the preceding steps have been completed successfully, an additional message is issued by the CA-7 CAIRIM initialization module:

CAL2002I CA-7 RIM: CA-7 TEST SYSTEM INITIALIZATION COMPLETE (SSCT=UCT7)

8. To run the test copy you need separate PROCs and CA-7 initialization file for the second copy of CA-7. You can use the members in the test version of JCLLIB created in the first step above. Run job CA07N020 from the test JCLLIB to copy them to a system or user PROCLIB.

Following is a summary of the differences between production and test CA-7 JCL procedures and options. If you specified TESTSYS=YES on the U7GEN macro, these procedures/files have already been set to run as TEST.

- a. Test CA-7 Online

On the SVCNO statement of the test CA-7 initialization file, add the parameter TEST=YES. **SVCNO,SASSVC=YES,TEST=YES** should be the entire statement.

Note: Whenever a job submitted from the test CA-7 requires CA-7 load processing, a parameter override of TEST=YES is automatically inserted with the LOAD step.

- b. Test CA-7 ICOM JCL Procedure

On the test ICOM PROC, set the TEST= PROC variable to TEST='TEST=YES'.

- c. Test CA-7 Trailer JCL Procedure

On the test TRAILER PROC (or through PROC variable override), set the TEST= PROC variable to TEST='TEST=YES;'. If you are executing a test trailer step without using a JCL procedure, add PARM='TEST=YES' to the SASSTRLR EXEC statement.

- d. Test CA-7 U7SVC JCL Procedure

On the test U7SVC PROC (or through PROC variable override), set the TEST= PROC variable to TEST='TEST=YES;'. If you are executing a test U7SVC step without using a JCL procedure, add PARM='TEST=YES' to the U7SVC EXEC statement.

- e. Test CA-7 Batch Card Load Program

If you wish to use the CA-7 Batch Card Load program (SASSBCLP) to run under the test CA-7, you must add a 'TEST=YES' parameter to the EXEC statement PARM= string.

- f. Test U7TESTSV Program

If you wish to use U7TESTSV program to run under the test CA-7, add PARM='TEST=YES' to the U7TESTSV EXEC statement.

- g. If you have any user-coded routines or jobs which reference the following JCL procedures or programs, ensure the TEST=YES parameter is specified on all calls which are to communicate with the test copy of CA-7.

PROC name	Program executed
CA7ICOM	SASSICOM
CA7LOAD	SASSJCL
CA7TRLR	SASSTRLR
CA7SVC	U7SVC
(none)	U7TESTSV
(none)	SASSBCLP

4.3 CA-7 Test Copy Limitations

The following are some test copy limitations:

1. Tracking pre-execution JCL Errors (can enable with ICMDSECT Zap - see 4-9)
2. Cannot track external Jobs/STCs
3. Cannot track external Data Sets

Chapter 5. Initialization

This chapter describes the initialization file structure, its rules of syntax and the initialization statements, keywords, and keyword values.

5.1 Syntax Rules

5.1.1 Initialization File Statements

Initialization file statements include a statement name followed by keywords and keyword values. Statement names (RESIDENT, INIT, SVCNO, and so forth) must begin in column 1. An asterisk in column 1 indicates a comment statement.

You must separate the first statement keyword from the statement name by a single comma (no blanks are allowed). Statement keywords are nonpositional unless otherwise noted. You must separate keyword values from subsequent statement keywords with a single comma (no blanks are allowed).

See Reading Syntax Diagrams in the *CA-7 Commands Guide* for more information.

5.1.1.1 Sample Statement Format

COL 1 name,keyword1=value,keyword2=value
--

Statements may not be coded beyond column 71. Blank statements are not allowed.

Statement continuation is accomplished by either:

1. coding values through column 71 and a nonblank character in column 72, or
2. leaving at least one blank after any comma.

Continuation then resumes with the first nonblank character in the next record which is not a comment statement. Continuation may begin anywhere before column 72.

Column 72 is reserved to optionally contain a nonblank character to indicate a statement is being continued. A nonblank character in column 72 is required only if a value is to be continued in the next record after breaking after column 71.

You may code comments in individual statements following at least one blank, but must not extend beyond column 71.

5.1.1.2 Sample Statement Continuation Formats

COL 1 	COL 72
name,keyword1=value,keyword2=value,keyword3=value,	*
keyword4=value	
name,keyword1=value,keyword2=value,keyword3=value,ke*	
yword4=value	
name,keyword1=value,keyword2=value,keyword3=value,	
keyword4=value	

Note: Continuation requires that at least one keyword is coded on the first line of the statement. Statements cannot be continued without at least one keyword on the first line of the statement.

5.2 Initialization File Statements and Keywords

The initialization file is input during activation of CA-7 and is contained in the data set defined by the UCC7IN DD statement in the CA-7 execution JCL. The file may be kept in a sequential data set or as a member of a partitioned data set. It may prove convenient to maintain different copies of the initialization file depending on requirements to alter the CA-7 processing configuration.

The CA-7 installation process generates sample CA-7 execution JCL and sample initialization files for batch and online initialization and places them in the CA-7 JCLLIB data set created during CA-7 SYSGEN. Generated Stage II JCL and initialization files require modification to suit installation requirements.

Chapter 6, “Execution” on page 6-1 provides samples of online and batch mode initialization files. Those examples may be helpful, as a reference, when reviewing the discussions in this chapter.

Depending on installation requirements and on configuration requirements during a given CA-7 execution, you may omit some initialization statements. Other statements must always be included, and some statements may need to be specified multiple times.

Statements in the initialization file define:

- The host operating system and job entry subsystem
- The number of CPUs in the CA-7 configuration
- The type of initialization to be performed
- The CA-7 terminal network
- Workstations and the physical terminal each is assigned
- User JCL data sets
- CA-7 modules to be available

Table 5-1 on page 5-5 is a matrix of the initialization statements and keywords. When used, statements must appear in the file in the sequence shown in Table 5-1 on page 5-5. The letter R identifies required statements. The letter M identifies those which may appear multiple times within the file. The letter C identifies statements which you may continue from one statement or line to the next.

Table 5-1 (Page 1 of 2). Initialization Statement Sequence				
Statement	Required	May code multiple times	Possible continuation necessary	
RESIDENT	R		C	
CUST	R			
NEWS		M		
INIT	R		C	
SVCNO	R			
APPLCTN	R	M		note 5
FMTBLK	R	M		note 5
CALBLK	R	M		note 5
UCC7VTAM			C	
GROUP	R	M	C	
LINE	R	M	C	
TERM	R	M	C	
INIT2				
SECURITY	R			note 7
STATIONS	R	M	C	
STNCAL		M	C	
DBASE	R		C	
ALOG1	R			
ALOG2				note 1
RESTART			C	note 2
SCHEDULE			C	note 3
CPU	R	M	C	
CALENDAR				
JCL	R	M	C	
JCLCHECK				note 4
FORMAT	R	M		
DAIO	R			
NETMAN				note 6
OPTIONS		M		

Table 5-1 (Page 2 of 2). Initialization Statement Sequence				
Statement	Required	May code multiple times	Possible continuation necessary	
VRMOPTS		M		
PFTERM		M		
PFnn/PAnn		M		
CANCEL				
END				
<p>Note 1: If used, ALOG1 must also be present.</p> <p>Note 2: Required only if restart reasons are to be required of the operator or CA-11 interface is desired.</p> <p>Note 3: Required whenever CA-7 is initialized for automatic scheduling control.</p> <p>Note 4: Required only if the CA-7 CA-JCLCheck interface is desired.</p> <p>Note 5: Associated load module required.</p> <p>Note 6: Required with the realtime CA-Netman interface.</p> <p>Note 7: Described in the <i>CA-7 Security Guide</i>.</p>				

The statements presented in Table 5-2 are the only allowable statements in the initialization file. This topic gives a detailed explanation of each statement and its associated keywords. All statements must begin in column 1.

The following statement descriptions appear in the order in which they appear in the initialization file.

Table 5-2 (Page 1 of 3). Initialization File Statements and Keywords			
Statement	Keywords	Required	Optional
RESIDENT	APGPL, NETWORK	X	
	JCLDEFS		X
CUST	ID	X	
NEWS	M		X
INIT	CONFIG, TYPE	X	
	RUNOP, TIMLIM, PERFORM, VALIDATE		X
SVCNO	SASSVC	X	
	DORMENQ, DORMVER, RNAME, MONITOR, ROUTER, TEST, XPSSCHD		X
APPLCTN	NAME	X	
	ATTR		X
FMTBLK	NAME	X	
	ATTR		X
CALBLK	NAME	X	
	ATTR		X
UCC7VTAM	APPL	X	
	OPEN, NUMRECV, VTNAME		X
GROUP	DEVICE, LNAME, NAME	X	
	OPEN, VLOGOFF		X
LINE	BUFSIZE, NAME, TNAME	X	
	OPEN		X
TERM	DEVICE, LINLEN, NAME, NLINE	X	
	CONS, LCLSNA, LOGMODE, MONLIM, PRINTER, RELEASE, SIMLOG, TIMLIM, VLOGOFF, VTAMID		X
INIT2			X

Table 5-2 (Page 2 of 3). Initialization File Statements and Keywords			
Statement	Keywords	Required	Optional
SECURITY	See the <i>CA-7 Security Guide</i> for a complete discussion of the SECURITY statement and its parameters.	X	
STATIONS	TRMID, STANIDS	X	
STNCAL	STN	X	
	CAL, FROM, TO		X
DBASE		X	
	JOB, LDJOBNM, DEFAULTJOB, DEFAULTTPS, LOADDSNS, PROCLOAD, RLOGDAYS, RSRC		X
	X		
ALOG1	DSN	X	
ALOG2	DSN	X	
RESTART	ARF, REASON, RMS, PROCRMS, STEPRMS,		X
	PARMRMS, WTO, WTOSTEP, ROUTCDE, LRTCDREQ		X
SCHEDULE	HIJBNUM, QDWELL, REPROMPT, SCNINCR, SCNSPAN	X	
	ABR, ENDDAY, PSFORCE, QPRG, REQUE, RETRY, SCHID, STOPQ		X
CPU	BARRIER, DDNAME, HOST, MAINIDS	X	
CALENDAR	DSN		X
JCL	DSN, INDEX	X	
	DPROC, DSORG, LTERM, DYN, ALT, MCD		X
JCLCHECK	DDNAME, MAXUSERS, TCBSEC		X
FORMAT	DDNAME, FLNAME	X	
	DISP, INDEX		X
DAIO	DASD	X	
	SWFTRK, SCRTRK, BUFSIZE, IOBS, RECORDS		X
NETMAN	DBASENM	X	
	EXIT, GOODCOMP		X
OPTIONS	ATTACH, AUTOREQ, CMGRDELAY, CPM, DFLTWLB, DPROCCOM, EXPDTCHK, EXTSCHID, FCMAXLEV, INITCASE, JOBDEL, JSOP, LATEPROMPT, LOADCLASS, OVJCL, PERMDSN, PROPDSNU, PROPMAIN, RLOGBDEF, RUNCLASS, SHUTDOWN, VRMDD, XQOFF		X

Table 5-2 (Page 3 of 3). Initialization File Statements and Keywords			
Statement	Keywords	Required	Optional
VRMOPTS	RPLCNT		X
PFTERM	TERM/T		X
PFnn/PAnn			X
	PFnn(nn=01-24)		X
	PAnn(nn=01-03)		X
CANCEL	Reason		X
END			X

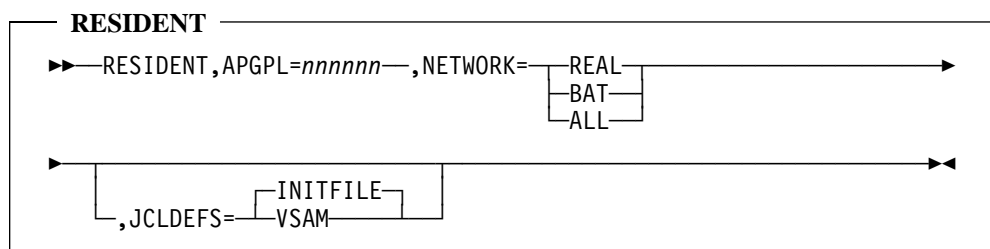
5.2.1 RESIDENT Statement

The RESIDENT statement defines the type and quantity of certain resources used while CA-7 is active.

- The types of terminals.
- The size of the CA-7 application pool and buffer pools.
- The source of the CA-7 JCL statements.

The RESIDENT statement is required. You may specify it only once, but you may continue it. All keywords are nonpositional and may appear in any order.

5.2.1.1 Syntax



Where:

APGPL

Specifies the memory size of the CA-7 application pool which is used for application programs not designated as resident. Value must be given as number of bytes (left-justified). APGPL is required and has no default. The size of the application buffer pool may affect CA-7 system performance. The value should not be less than 120000. Any amount over 240000 is probably wasted space.

See the /DISPLAY command in the *CA-7 Commands Guide* for a discussion of monitoring pool sizes through the /DISPLAY,POOLS=ALL command.

NETWORK

Defines the type of terminals included in the CA-7 terminal network. NETWORK is required and has no defaults. For CA-7 online execution, NETWORK=ALL should be specified. Value must be one of the following:

REAL

Only real terminals (3270s and associated printers) and the OS system console (optional) are defined. The batch terminal interface facility cannot be used.

BAT

Only batch terminals are defined.

ALL

Real terminals, batch terminals, and the OS system console (optional) are all defined.

JCLDEFS

Specifies the source of the CA-7 JCL library definitions that use symbolic indexes. Any changes to JCL library definitions made with the /JCL command are stored on the CA-7 database. The JCLDEFS option determines how these stored definitions are used when CA-7 is initialized.

INITFILE

This is the default. Erase any JCL library definitions found on the database. Next, read the JCL library definitions in the CA-7 initialization file. Store all the definitions using symbolic indexes in the database.

VSAM

When CA-7 is initialized, all JCL library definitions that use symbolic indexes are read from the database. Any definitions using symbolic indexes found in the initialization file are ignored.

Note: This option does not affect JCL library definitions that use a numeric index (such as INDEX=10). These definitions **must** reside in the CA-7 initialization file to be recognized.

5.2.2 CUST Statement

The CUST statement specifies a heading shown on logo screens and used by history and management reporting facilities to customize user reports and graphs. This normally contains the company name supplied by the user. This statement must immediately follow the RESIDENT statement and may not be omitted. Continuation is never needed. Comments are not allowed at the end of this statement. CA-7 considers all text entered as part of the ID.

5.2.2.1 Syntax

<p>CUST</p> <p>▶▶—CUST,ID=xx...x—▶▶</p>
--

Where:

ID

Specifies the heading value to appear on CA-7 logo screens, reports and APA graphs. Value may be any string of characters, up to 44, that the user desires. At least one nonblank is required.

5.2.3 NEWS Statement

The NEWS statement specifies message text which can be used to broadcast a free-form message to users who sign on to CA-7. The NEWS statement can be specified twice in the initialization file to allow for two lines of message data to appear on the Logon screen. The message(s) only appear on the Logon screen. You must place this statement in the initialization file after the CUST statement.

5.2.3.1 Syntax

<p>NEWS</p> <p>▶▶—NEWS,M=xx...x—▶▶</p>

Where:

M

Specifies the text of the message data to appear on the Logon screen. Value may be any string of characters, up to 65 characters in length, as desired.

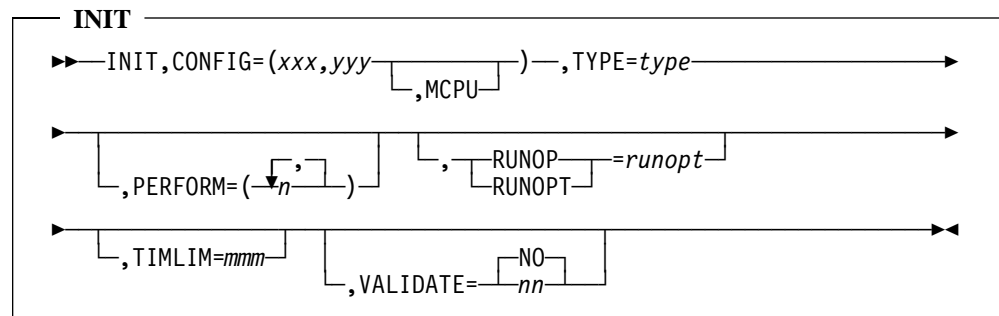
5.2.4 INIT Statement

The INIT statement specifies the initialization options to be used when activating CA-7 and the hardware/software configuration of the host environment on which CA-7 is active. The options are:

- The type of initialization processing to be done.
- Run options to be in effect for CA-7.
- The operating system, job entry subsystem, and CPU configuration.
- Whether the operator is to validate the system date and time of day at startup time.

The INIT statement must always be present. You may specify it only once, and you may continue it. All keywords are nonpositional and may appear in any order following INIT.

5.2.4.1 Syntax



Where:

CONFIG

Specifies the operating system, job entry subsystem, and CPU configuration of the host environment on which CA-7 is active. Multiple options may be specified as a sublist, enclosed with parentheses and separated by commas. CONFIG is required and has no defaults. Value is given as a subparameter list. Values must be as follows:

xxx

Identifies the operating system. Values can be VS2, MVS, or SVS.

yyyy

Identifies the job entry subsystem. Values can be JES2 or JES3.

MCPU

Indicates multiple CPUs. Value can only be MCPU. In a single CPU environment, MCPU should be omitted. Currently used only for documentation purposes.

TYPE

Specifies the type of initialization process to perform. TYPE is required and has no default. Any TYPE specification parameter on the EXEC statement in the CA-7 JCL overrides this TYPE parameter; however, you must include one here in all cases. This parameter is discussed in detail in 6.3.1, “Startup Options” on page 6-18. Briefly, the value must be one of the following:

WARM

This value is used to reactivate CA-7 after a normal shutdown and if no environmental changes are made.

ERST

This value is used to restart CA-7 after certain system failures or after certain initialization file changes.

COLD

This value is used to reactivate CA-7 after certain system failures which are not totally recoverable with ERST. It can also be used to simply purge the queues.

MOVQ

This value is used to reactivate CA-7, following a shutdown which unloaded the queues, to reload the queues before continuing.

FORM

This value is used to activate CA-7 the first time after installation or in some cases where new queues are to be used.

DORM

This value is used to start a dormant copy. See Chapter 6, “Execution” on page 6-1 for more information on this option.

EDIT

Used to verify the initialization file. See 5.3, “Offline Verification of Initialization File” on page 5-86 for more information.

Note: The TYPE= in the PARM in the CA-7 PROC overrides whatever is on the INIT statement TYPE= keyword.

PERFORM

Indicates the performance options required for this execution of CA-7. PERFORM is optional. Value can be one or more single character numbers, enclosed in parentheses and separated by commas, as follows:

1

Skip SMF type 14 records. This reduces database access during the SMF feedback process, and suppresses messages about data sets not used.

2

Not used.

3

Not used.

4

Not used.

5

Skip duplicate checking for queue adds. This reduces EXCPs on the queue data sets.

6

Skip DSORG=PO data set feedback. This reduces SMF task usage.

7Cause CA-7 to use dynamic allocation instead of OPEN TYPE=J. This option does not require the use of U7volser DD statements. If there are no U7volser DD statements in the CA-7 JCL, option 7 is assumed. If there is at least one U7volser DD statement in the CA-7 JCL, then 7 or 8 (or both) must be specified to cause CA-7 to use dynamic allocation.**8**

Causes CA-7 to use dynamic allocation instead of OPEN TYPE=J. The use of 8 without specifying 7 indicates that the U7volser DD statements in the CA-7 JCL are required although dynamic allocation is used for the data sets. If there is at least one U7volser DD statement in the CA-7 JCL, then 7 or 8 (or both) must be specified to cause CA-7 to use dynamic allocation.

RUNOP|RUNOPT

Indicates the specific run options active when initialization is complete. Multiple options may be specified as a sublist, enclosed within parentheses and separated by commas. RUNOP(T) is optional. If REPT and MAIN are both omitted, an online run is assumed. MAIN and REPT should not be used for online execution. The default is NSTA, but the value may be one or more of the following:

SCAN

Indicates that schedule scan is to begin as soon as initialization is complete. SCAN may not be specified with REPT or MAIN. Should be specified to engage automatic scheduling functions.

NSTA

This is the default. However, NSTA is currently a dummy entry, meaning that schedule scan is not active when CA-7 initializes.

MAIN

Indicates a maintenance only run. Information from the communications data set is used to update the database and queues. No scheduling, job submission, or SMF monitoring is done. If MAIN is specified, REPT and SCAN must be omitted.

NCWT

Indicates no wait is allowed if a GETMAIN for additional core fails. CA-7 abends. NCWT should not be specified for online runs.

REPT

New JCL may be added, database maintenance may be performed and reports may be run. No scheduling, job submission, SMF monitoring, or SMF updating is done. If REPT is specified, MAIN and SCAN must be omitted.

TIMLIM

Defines an elapsed time limit for CA-7 execution. When the limit is reached, CA-7 automatically shuts down. Value must be given as a number of minutes. TIMLIM is optional for batch only type runs and should not be specified for online processing. If omitted, no time limit is imposed for online runs, but 60 minutes is used for batch only type runs.

VALIDATE

Specifies whether the system date and time are to be validated by the operator at CA-7 startup. It allows the user to specify that operator validation is not to be performed; or to define a time tolerance for disabling the operator validation process. The date/time tolerance defines the allowable difference between the current system date/time values and the latest CA-7 checkpoint date/time values. If the date/time values are within the time tolerance, operator validation is not performed. VALIDATE is an optional parameter. If used the values must be one of the following:

NO

Indicates that date/time validation is not to be performed by the operator. This is the default value.

nn

Numeric value from 1 to 99 indicating the allowable time tolerance in hours. A value of 0 (zero) indicates that date/time validation must always be performed by the operator regardless of the date and time values in either the system clock or the checkpoint record.

For WARM or ERST starts with VALIDATE=0 or date and time values outside the tolerance, the last checkpoint and last schedule scan date/time values are displayed through a WTO.

For all other conditions where verification is to be performed, the current system clock date and time values are then displayed followed by a WTOR requesting operator action. If necessary, the system date and time should be corrected on all CPUs before responding to the WTOR. Response to the WTOR should be one of the following:

U Indicates initialization may proceed normally.

NOSCAN Initialization may proceed but schedule scan is disabled.

Following a response of U or NOSCAN, the current system clock date and time values are displayed again before continuing. This allows the user to verify any adjustments made while the WTOR was outstanding. The results of any adjustment should be verifiable from the before and after displays.

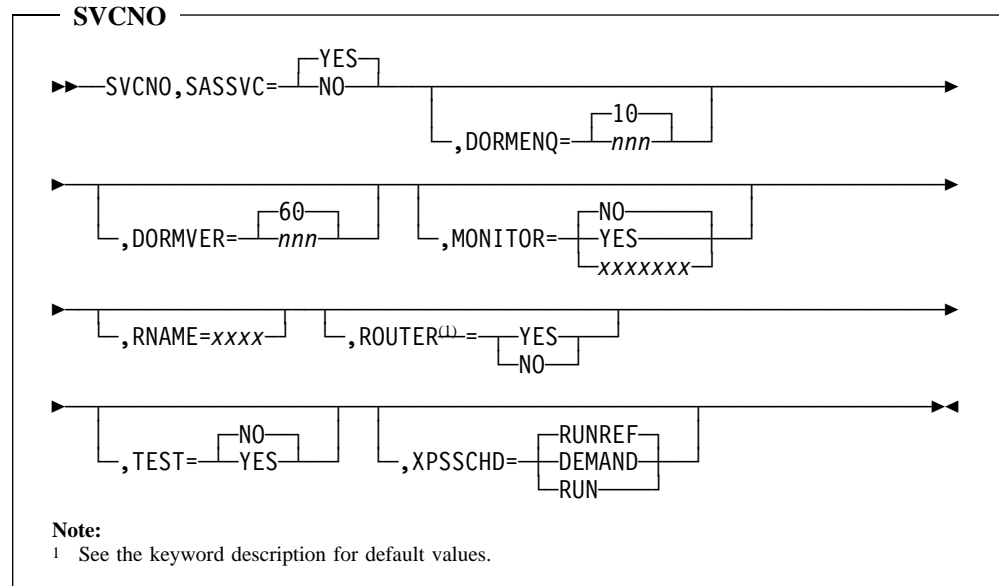
Whenever nonzero time tolerance values are used, the date and time are validated for only the CPU on which CA-7 is running. The operator should validate the system clock date and time values on all CPUs before responding to the WTOR.

For further discussion of this option, see 6.2, "Initialization Considerations" on page 6-15.

5.2.5 SVCNO Statement

The SVCNO statement indicates the availability of the Type IV SVC assigned to CA-7. This statement must always be present and may only be specified once. Continuation is never necessary.

5.2.5.1 Syntax



Where:

SASSVC

Indicates whether the SVC is available. SASSVC is required. Value may be YES or NO.

YES

CA-7 obtains the SVC number from the ICMDSECT module addressed by the SSCT. This is the default.

NO

CA-7 executes without the SVC number. This should only be used for testing purposes when job tracking is not desired.

SASSVC=NO can be used for installation testing of CA-7. No jobs can be tracked, but the various formatted screens can be displayed and functions performed, such as adding jobs, schedules, user documentation, and so forth. The editor can also be used. When testing in this manner, RUNOPT=REPT should be specified in the INIT statement.

DORMENQ

Specifies the number of seconds that a dormant copy of CA-7 will wait between checks for an inactive production copy of CA-7.

nnn

The number of seconds between checks. The default is 10.

DORMVER

Specifies the number of seconds that a dormant copy of CA-7 will wait, after it determines that the production is inactive, before completing initialization and becoming the production copy of CA-7.

nnn

The number of seconds to wait. The default is 60.

MONITOR

Identifies this copy of CA-7 as an XPS SUBMIT MONITOR that can receive cross-platform scheduling requests.

Allowable values are YES, NO, or a 7 byte monitor name that will be used. The default value is NO. If the value of MONITOR is YES, a monitor name is generated. If SASSVC=NO, the MONITOR keyword is ignored.

NO

CA-7 does not receive cross-platform scheduling requests. This is the default.

YES

A monitor name consisting of a string of 7 nonblank or null characters will be generated. The resulting string begins with CA7 if the production CA-7 SVC is used (TEST=NO is specified or taken by default). If the test CA-7 SVC is used (TEST=YES is specified), the string begins with TS7. The remaining 4 positions of the string are filled with the SMF ID where CA-7 executes. Any blank positions at the end of the string will be filled with an EBCDIC 'X'.

Examples: Suppose MONITOR=YES is coded and the production CA-7 SVC is used. The SMF ID where CA-7 runs is MVS1. The generated monitor name is CA7MVS1. If the SMF ID is ST1, the generated name is CA7ST1X. If TEST=YES is coded on the SVCNO statement and the SMF ID where CA-7 runs is MVS, the generated monitor name is TS7MVSX.

xxxxxxx

The monitor name to be used may be declared as a value of the keyword. It must consist of 7 nonblank or null EBCDIC characters.

Example: MONITOR=XXX is not a valid specification because it is not long enough. MONITOR=CA7 1 is not valid because a blank is embedded in the keyword value and because it is not long enough.

Note: You must ensure that the monitor name generated or specified is unique within the domain of the XPS ROUTER. If it is not unique, the XPS SUBMIT MONITOR function terminates with an error.

A console message during CA-7 initialization announces the value for MONITOR that is generated or specified.

RNAME

This is used to change the RNAME used to ENQ ICMDSECT at startup. The default is the subsystem name. For example, UC07 is the default subsystem name for a production copy of CA-7, and UCT7 is the default subsystem name for a test copy of CA-7. The xxxx value can be 1 to 4 alphanumeric characters.

ROUTER

Indicates whether the XPS SUBMIT ROUTER will run in this CA-7 address space.

YES

Indicates that the XPS ROUTER will run in this address space.

NO

Indicates that the XPS ROUTER will not run in this address space.

The default value for this keyword depends on the values of TEST and MONITOR.

If TEST=YES is specified, the default value of ROUTER is NO. It is assumed that the XPS ROUTER will run in the address space associated with the production copy of CA-7.

If MONITOR=NO is specified, the default value for ROUTER is also NO.

If neither of the above conditions is true, the default value for ROUTER is YES.

TEST

Indicates whether to use production or test CA-7 SVC.

NO

The production CA-7 SVC is used.

YES

The test CA-7 SVC is used.

XPSSCHD

Identifies the CA-7 command to be used for cross-platform scheduling requests. When CA-7 receives such a request, a CA-7 command to schedule the job is built using parameters supplied by the requester. The allowable values are RUNREF, DEMAND, and RUN.

XPSSCHD=RUNREF is the default.

RUNREF

Specifies that the job requested does not trigger other jobs, it does not wait for satisfaction of requirements, nor does it satisfy requirements for other jobs. Such a job cannot be restarted in CA-7, but is considered complete when CA-7 is notified of its completion on MVS. A job that is requested by an XPS CLIENT when this option is in effect reports an entry mode of 'XPS' in the output of queue display commands such as LQ.

Because of the restrictions on requirements and restart, jobs requested using this option encounter fewer conditions preventing immediate submission and cause notification of completion to be sent to the requester in a timely manner.

Use of this option minimizes processing delays due to conditions which might otherwise require manual intervention.

If this option is used, triggers, requirements, and job rerun conditions should be defined and documented in the XPS CLIENT. This ensures that the primary responsibility for workload control is assigned to the XPS CLIENT (typically the Unicenter TNG Workload Manager).

However, if it is important that CA-7 retain restart control or if it is important that CA-7 requirement and trigger definitions be honored, you should consider the other XPSSCHD options.

DEMAND

Indicates that the job named on the cross-platform scheduling request is scheduled using the DEMAND command. CA-7 verifies the availability of input requirements for the job prior to submission and updates the database on successful completion of the job. Triggers defined for the job are honored when the job successfully completes. A job scheduled with this option exits the request queue only when the job completes normally. Otherwise, it remains in the request queue to be restarted. See the *CA-7 Commands Guide* for additional information on the DEMAND command. The entry mode for a job requested by an XPS CLIENT when this option is used is XDEM in the output of queue display commands such as LQ.

RUN

Indicates that the job named on the cross-platform scheduling request is scheduled using the RUN command. CA-7 does not verify requirement availability for the job. This instance of the job does not satisfy any requirement other jobs may have for this one. Triggers defined for this job are not honored for this instance of the job. However, like the job scheduled with DEMAND, a job scheduled using RUN exits the request queue only on normal job completion. Thus, the job can be restarted in CA-7. See the *CA-7 Commands Guide* for additional information on the RUN command. The entry mode for a job requested by an XPS CLIENT when this option is used is reported as XRUN in the output of display commands such as LQ.

Note: It is important to remember the XPS CLIENT is notified of job completion only when the job requested exits the request queue. If XPSSCHD=RUNREF, this occurs when CA-7 is notified of job completion (whether normal or abnormal). If XPSSCHD=DEMAND or XPSSCHD=RUN, this occurs when the job completes normally in CA-7, or when the job is forced completed, or if the job is canceled in CA-7 prior to submission.

If cross-platform scheduling is used, give careful consideration to the assignment of responsibilities for monitoring and controlling the workload. The value on the XPSSCHD parameter is important because it affects distribution of those responsibilities. To determine the appropriate value for this parameter, you must decide if the greatest degree of control should be given to the XPS CLIENT (typically Unicenter TNG Workload Manager) or if the XPS SERVER (in this case CA-7) should share responsibility for such control.

If the primary point of workload control is to be the XPS CLIENT, then XPSSCHD=RUNREF is recommended. In this case triggers and requirements should be defined in the XPS CLIENT. Jobs cannot be restarted, but may be rerun from the XPS CLIENT.

If a greater degree of control for the XPS SERVER is desired, XPSSCHD=DEMAND or XPSSCHD=RUN may be used. If you use one of these options, you may use the CA-7 restart facilities for the job. If you use XPSSCHD=DEMAND, CA-7 trigger and requirement definitions are honored. Although these options allow use of CA-7 controls over the workload on the XPS SERVER, there is a greater likelihood for processing delays due to requirement posting or restarts that cannot be directly controlled from the XPS CLIENT.

Note: You may override this value for selected jobs by adding the XPSSCHD keyword after the job name in the 'filename' field (Job detail - submission - filename). See the *CA-7 Interfaces Guide*, CA-7 XPS SERVER Implementation Checklist, for additional information.

5.2.6 APPLCTN Statement

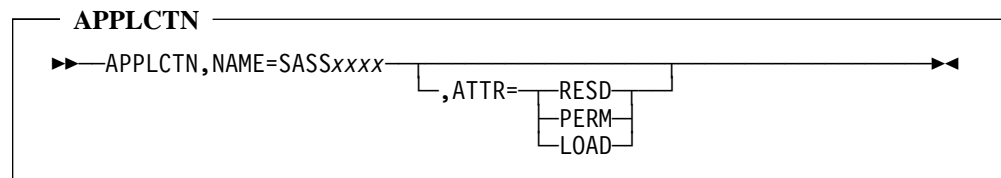
APPLCTN statements define all CA-7 application modules that are available for use after initialization is complete. It also indicates modules which must be CA-7 resident and modules which require resources for input/output activity.

All of the supplied CA-7 application program modules are defined in the SASSPROG module. The APPLCTN statement included in the skeleton initialization file refers to this load module and has the following format:

APPLCTN,NAME=SASSPROG,ATTR=LOAD

Additions to the initialization file are made by inserting additional APPLCTN statements before SASSPROG in the initialization file. This is only needed for interface modules or to mark modules as PERM or RESD, if necessary.

5.2.6.1 Syntax



Where:

NAME

Specifies the name of a CA-7 program module. Value must be the 8-character program name. The first 4 characters must be SASS. The last 4 identify the CA-7 application and the specific module within that application. NAME is required and has no default. It must be specified before the other keywords.

If ATTR=LOAD, then NAME=SASSxxxx is the name of a module created from APPLCTN macros (other modules to be loaded). It is recommended that only SASSPROG be used with the ATTR=LOAD keyword.

ATTR

Specifies where a module is to be loaded. ATTR is optional. If ATTR is not coded, the specified module is loaded in the application pool (APGPL) by CA-7 when it is required. If ATTR=RESO or PERM is entered, the module is loaded during initialization of CA-7. If used, the value must be one of the following:

RES**D**

The module is made permanently resident within the memory allocated to the CA-7 resident pool.

PERM

The module is loaded in the CA-7 job pack area by an OS LOAD macro. Memory for these modules should be included within the value for the CWORK keyword in the PARM statement of the CA-7 EXEC statement.

LOAD

Indicates NAME is a load module created from APPLCTN macros. If **ATTR=(LOAD,PERM)** is used, each module specified by NAME is marked PERM.

5.2.7 FMTBLK Statement

The FMTBLK statement identifies all format block modules used for message input and output by the CA-7 applications.

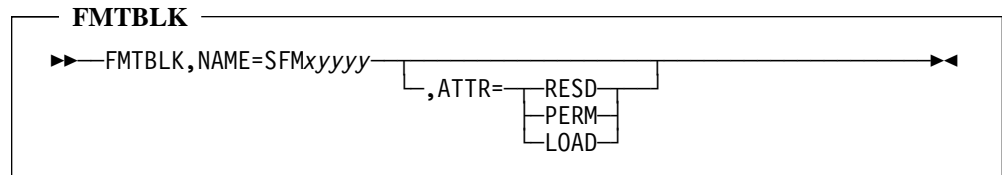
As with APPLCTN statements, the FMTBLK statements for all supplied CA-7 format blocks are defined in two load modules. These are SASSFMTH, the load module containing all batch FMTBLK statements, and SASSFMTH, the load module containing all online FMTBLK statements. Two FMTBLK statements are included in the skeleton initialization file that refer to these load modules. They have the following format:

```
FMTBLK,NAME=SASSFMTH,ATTR=LOAD
FMTBLK,NAME=SASSFMTH,ATTR=LOAD
```

Additions to the initialization file are made by inserting additional FMTBLK statements before those included in the skeleton initialization file. A continuation should never be necessary. The optional keyword (ATTR) must not be specified before FMTBLK or NAME.

5.2.7.1 Syntax

This is the format when adding FMTBLK statements.



Where:

NAME

Specifies the name of a format block module. NAME is required and has no default. Value must be in the form SFMxyyyy, as follows:

x

Indicates the device class for which the module is used. Online devices require an H. Batch type devices require an A.

yyyy

Indicates the CA-7 application program(s) to which the format block belongs.

ATTR

Specifies how the format block is to be loaded. ATTR is optional. If not specified, the format block is loaded by CA-7 as needed for terminal input and output. If ATTR=RESD or PERM, the format block is loaded during CA-7 initialization. If used, the value must be one of the following:

RESD

The format block is made permanently resident within the memory allocated to the CA-7 resident pool.

PERM

The format block is loaded in the CA-7 job pack area by an OS LOAD macro.

LOAD

If ATTR=LOAD is used, the NAME is the name of a load module created from FMTBLK macros. In this case, the NAME must be of the form SASSxxxx.

It is recommended that ATTR=LOAD only be used with the supplied initialization file statements that specify SASSFMTH and SASSFMTA. It should not be necessary to add FMTBLK statements. The two that are supplied should be sufficient.

5.2.8 CALBLK Statement

The CALBLK statement identifies the base calendars defined by the user with the CALENDAR macro. At least one base calendar must be defined before job and/or network schedules can be resolved.

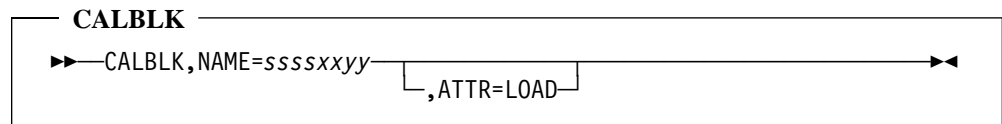
The CA-7 SYSGEN generates JCL to create sample base calendars that are referenced in the skeleton initialization file by CALBLK statements.

Additions to the initialization file are made by inserting additional CALBLK statements or replacing the sample CALBLK statements in the skeleton initialization file.

A CALBLK statement must be specified for each active base calendar if no CALENDAR statement is coded. Continuation of a CALBLK statement should never be necessary.

5.2.8.1 Syntax

This is the format when adding a CALBLK statement.



Where:

NAME

Specifies the name of a base calendar or a module containing base calendar definitions. NAME is required and has no default. If NAME specifies an individual calendar, the ssss must be SCAL. If it specifies a module defining base calendars, use SASS as the first 4 characters followed by any other 4 characters. If it is an individual calendar (SCAL) the xxyy values are:

xx

Identifies the year being defined by the calendar (00, 01, and so forth). The xx value must correspond to the value specified for YEAR on the CALENDAR macro statement.

yy

Identifies the unique name assigned by the user at the time the base calendar was generated. The yy value must correspond to the value specified for SCAL on the CALENDAR macro statement.

ATTR

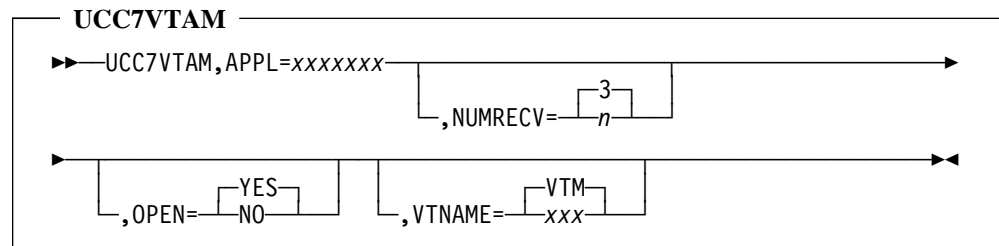
Indicates NAME is a load module that defines base calendars using the CALBLK macros. ATTR is optional. If used, value must be LOAD. If omitted, the CALBLK statement specifies the name of an individual calendar.

5.2.9 UCC7VTAM Statement

The UCC7VTAM statement is required if VTAM terminals are connected to CA-7 and defines two types of information to CA-7:

- Information to access the 3270-type terminals defined later in the initialization file with GROUP, LINE, and TERM statements (DEVICE=3270V).
- VTAM options used by CA-7.

5.2.9.1 Syntax



Where:

APPL

Specifies the name of the VTAM APPL macro that defines CA-7 to VTAM. APPL must be specified in up to 7 characters, is required and has no default.

NUMRECV

Specifies the number of VTAM RECEIVE macros to be outstanding concurrently in CA-7. This is the number of inputs from VTAM terminals which can be processed simultaneously. NUMRECV is optional. If omitted, the default is NUMRECV=3.

OPEN

Specifies if the CA-7 VTAM ACB is to be opened during initialization. OPEN is optional. If used, value must be one of the following:

YES

Opens the VTAM ACB during CA-7 initialization. CA-7 attempts to acquire those terminals that have SIMLOG=YES specified on their TERM statements. This is the default.

NO

The VTAM ACB is not opened during CA-7 initialization. A /OPEN,GROUP=(the first VTAM group in the initialization file) must be entered through a non-VTAM CA-7 terminal to enable VTAM terminals to access CA-7.

VTNAME

Specifies the first 3 characters of the CA-7 virtual terminal names. If VTNAME is specified, it must be 3 characters. If VTNAME is omitted, the default is VTM. See 5.2.11, “LINE Statement” on page 5-31 for additional details on virtual terminal usage by CA-7.

5.2.10 GROUP Statement

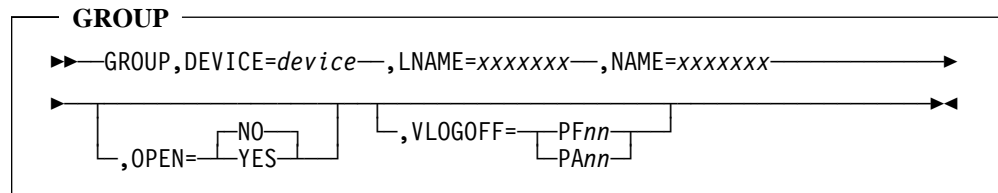
The GROUP statement defines the name and characteristics of a logical line group to which one or more physical lines with terminals may be associated. Multiple line groups (and GROUP statements) are required when terminals with varying device classes are defined. GROUP statements are supplied with the skeleton initialization file to define several types of terminal groups. They may require modification to fit the user environment.

The line group is the top level of a hierarchical structure which includes groups, lines, and terminals. The GROUP statement must be followed by the LINE, and TERM statement(s) which identify the elements of that group. If multiple line groups are being defined, all the associated lines and terminals must be specified before the next GROUP statement defining a new line group is introduced. An example of the sequence of the GROUP, LINE, and TERM statements could be:

```
GROUP,...
LINE,...
TERM,...
TERM,...
GROUP,...
LINE,...
TERM,...
and so forth.
```

At least one GROUP statement is required to allow for definition of the CA-7 master terminal (for online initialization), or a batch terminal to be used for input and output (for batch or maintenance initialization). The GROUP statement may be continued.

5.2.10.1 Syntax



Where:

DEVICE

Specifies the symbolic device class of the terminal belonging to the line group. DEVICE is required and has no default. Value must be one of the following:

3270V

Indicates the line group contains 3270-type VTAM terminals and optionally, the associated printer. These terminals may be local or remote.

BATCH

Indicates the line group consists of a single batch terminal. No more than eight batch terminal groups (with DEVICE=BATCH) may be defined for each CA-7 initialization.

CCI

Indicates the line group consists of a single CCI terminal.

CONSL

Indicates the line group consists of the OS system console. Only one GROUP with DEVICE=CONSL may be specified per CA-7 initialization. When this is opened, communication is performed using the MODIFY command.

TRLDV

Indicates the line group consists of the logical terminal to be used for trailer step transactions. Only one GROUP with DEVICE=TRLDV may be specified per CA-7 initialization.

TRXDV

Indicates the line group consists of logical terminals dedicated to transactions scheduled internally by CA-7 functions. At least one such terminal must be defined if either of the following CA-7 functions are used:

- ARF (RESTART, ARF=YES is specified)
- CA-7 XPS SERVER support

BSAM

Indicates the line group consists of a single, print-image browse data set to receive the majority of the CA-7 messages. Only one group with DEVICE=BSAM may be specified.

LNAME

Specifies the name, up to 7 characters, of the line assigned to this line group. Value must be the line name(s) defined by the NAME parameter on the associated LINE statement(s). If multiple line statements are input after the GROUP statement, multiple line names must be specified here as a sublist separated by commas and enclosed in parentheses. LNAME is required and has no default.

NAME

Specifies the name, up to 7 characters, associated with the line group.

- For the browse data set, the name must be the same as the ddname used to define the data set in the CA-7 execution JCL.
- For batch terminals (DEVICE=BATCH), the NAME must correspond to the first 7 characters of the ddname in the CA-7 JCL which defines the input and output data set pair. A separate GROUP, LINE, and TERM statement is required for each batch terminal. This implies a separate pair of input and output DD statements for each batch terminal defined in the CA-7 JCL. The last character of the ddname in the CA-7 JCL is either an I for input data set or O for output data set.
- For VTAM terminals (DEVICE=3270V), only one GROUP and LINE statement is needed to define all VTAM terminals. This is because only one ACB is opened.

OPEN

Specifies whether the terminals belonging to this group are available for processing transactions (when initialization is complete) without manual intervention by the MTO. OPEN is optional. If used, value must be one of the following:

NO

None of the terminals in the line group is immediately available on completion of initialization. This is the required value when DEVICE is specified as BATCH or BSAM. For other DEVICE values, the MTO has to manually activate the lines in the group to make the terminals available. This is the default.

YES

All terminals in the line group are immediately available on completion of initialization. YES must be used if either DEVICE=TRLDV or TRXDV is specified.

VLOGOFF

Specifies the program function (PF) or program access (PA) key used to log off CA-7 and return to VTAM. The format for program function keys is PFnn where nn is a number from 01 to 24. The format for program access keys is PAnn where nn is a number from 01 to 03. A 2-digit number is required such as PA01; PA1 is invalid. VLOGOFF is optional and is used with DEVICE=3270V only. When omitted, a program function key or program access key is not used to return to VTAM unless VLOGOFF is specified with the TERM statement. When specified on the GROUP statement, the specified key becomes the default for all 3270V terminals whose TERM statement does not specify a VLOGOFF parameter. If not specified in either statement, the terminal operator must issue a /CLOSE command, with no parameters, to return to VTAM.

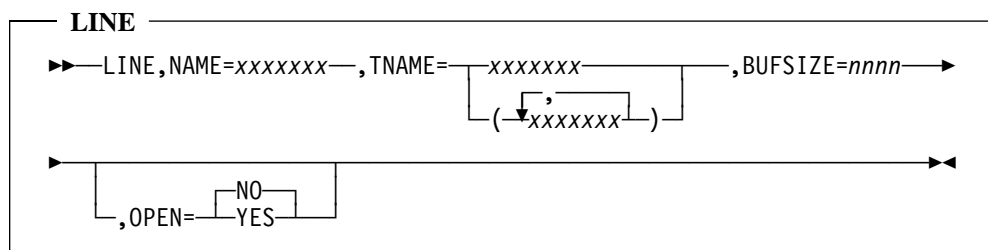
5.2.11 LINE Statement

The LINE statement corresponds to an I/O (buffer) area built by CA-7 during initialization.

At least one LINE statement is always required to allow for definition of the CA-7 master terminal (for online initialization), or the batch terminal used for input and output (for batch or maintenance initialization). In the case of virtual terminals, if there are multiple LINE statements, only one of them can use the virtual terminal special name defined with the TNAME parameter. The LINE statement may be continued.

All keywords are nonpositional and may appear in any order following LINE.

5.2.11.1 Syntax



Where:

NAME

Specifies the name, up to 7 characters, assigned to the line. Each LINE must have a unique name. Value must match an LNAME value (on the associated GROUP statement) which identifies this line. NAME is required and has no default.

TNAME

Specifies the name of the terminal(s). Value must be the name(s), up to 7 characters, designated by the NAME parameter on the TERM statement(s). Multiple terminal names may be specified as a sublist, enclosed in parentheses and separated by commas. For batch terminals, there may be only one terminal per line.

To use the virtual terminal feature for VTAM terminals, TNAME must begin with the 3 characters specified by VTNAME (see 5.2.9, “UCC7VTAM Statement” on page 5-27) followed by a #, followed by a 3-digit number. Leading zeros are required. If VTNAME= is not coded on the UCC7VTAM statement, the first three characters must be VTAM. This definition represents the maximum number of virtual terminals that can be used at the same time, with 255 being the maximum number. TNAME is required and has no default. ISPF interface

Note: You must define virtual terminals to use the CA-7 TSO/ISPF and CA-7 WorkStation interfaces.

You need to consider the size of the DQTQ data set when defining virtual terminals. See Chapter 3, “Installation Requirements” on page 3-1 for additional information.

BUFSIZE

Specifies the line buffer size in bytes. Value depends on the symbolic device type designated by **DEVICE** on the associated **GROUP** statement.

- If **DEVICE=BSAM**, the **BUFSIZE** value must be the block size of the browse data set.
- If **DEVICE=BATCH**, the **BUFSIZE** must be the block size of the batch input data set.
- If **DEVICE=3270x**, then **BUFSIZE** must be at least 3120.
- If **DEVICE=TRLDV**, then **BUFSIZE** must be 1024.
- If **DEVICE=CONSL**, then **BUFSIZE** must be 150.
- If **DEVICE=TRXDV**, then **BUFSIZE** must be 1024.

BUFSIZE is required for all **DEVICE** types except 3270V.

OPEN

Specifies whether the terminals on this line are available for processing transactions (as soon as initialization is complete) without manual intervention by the MTO.

OPEN is optional. If used, value must be one of the following:

NO

None of the terminals on this line is immediately available. This is the required value when **DEVICE** is specified as **BATCH** or **BSAM**. For online (3270x) device types, the MTO has to manually activate the line to make the terminals available. This is the default.

Note: If **OPEN=YES** is not specified for **DEVICE=CONSL**, then you will get the message **IEE324I MODIFY REJECTED - TASK BUSY** when trying to issue a modify command.

YES

All terminals on this line are immediately available on completion of initialization. **YES** must be used when **DEVICE=TRLDV** or **TRXDV** is specified.

5.2.12 TERM Statement

The TERM statement defines the terminals to be active after initialization. The type of terminal, its page structure and addresses are supplied by this statement. With the exception of virtual terminals, each terminal that is to have access to CA-7 must have a TERM statement. For real terminals, there may be multiple TERM statements associated with a single LINE statement.

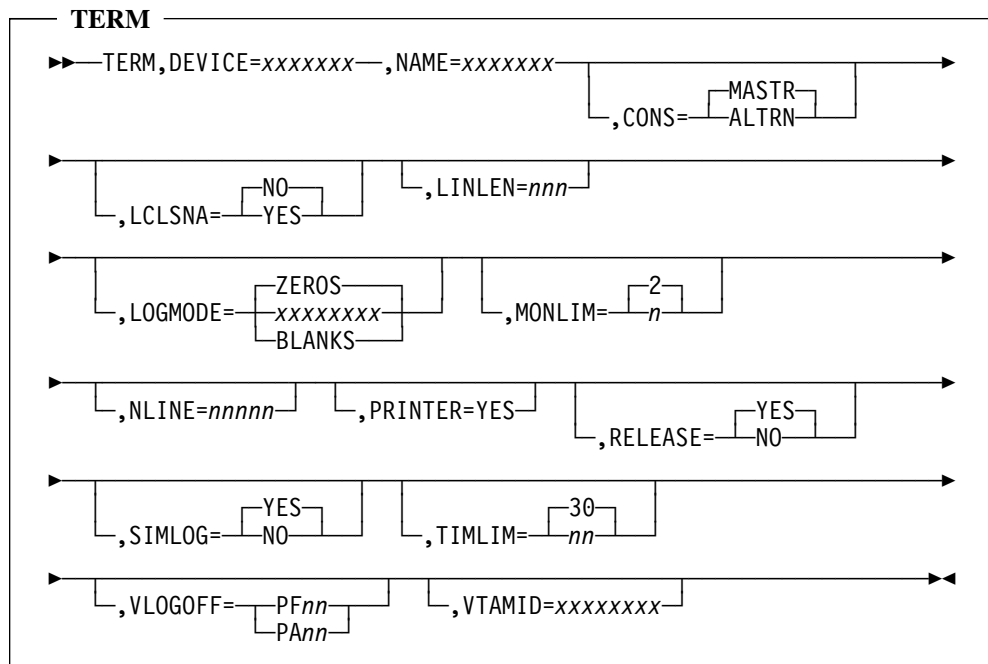
At least one TERM statement is always required to define either the CA-7 master terminal (for online initialization), or a batch terminal to be used for input and output (for batch or maintenance initialization). The TERM statement may be continued.

The theoretical maximum number of TERM statements, and therefore the maximum number of terminals, is 16,000. However, a more practical limit in the range of 100 to 200 terminals should be observed if possible. When the terminal network exceeds that size, memory requirements may become significant. Performance levels may prove to be less satisfactory due to competition between terminals for resources. To avoid potential performance problems, see 10.2, "Initialization File Considerations" on page 10-4 for a detailed discussion of performance considerations.

For the virtual terminal definition, a TERM statement is optional. If a TERM statement is not coded, CA-7 generates one TERM statement with a + in column 1 to indicate that terminal control blocks were built. Printers cannot be used as virtual terminals.

All keywords are nonpositional and may appear in any order following TERM.

5.2.12.1 Syntax



Where:

DEVICE

Specifies the symbolic device type of the terminal. **DEVICE** is required and has no default. Value must be one of the following and must match the **DEVICE** value given on the associated **GROUP** statement:

3270V

Indicates the terminal is a 3270-type device or associated printer, either local or remote, that is defined to VTAM.

BATCH

Indicates a batch terminal. Eight terminals with **DEVICE=BATCH** may be defined for each initialization of CA-7.

CCI

Indicates the terminal is used with the CA-7 CCI Terminal Interface.

CONSL

Indicates the terminal is the device designated as an OS system console. Only one terminal with **DEVICE=CONSL** may be defined for each initialization of CA-7. The OS console should never be defined as the master station (**STANIDS=MASTER** on the **STATIONS** statement).

TRLDV

Indicates the terminal to be used for input of trailer step commands. Only one terminal with **DEVICE=TRLDV** may be defined for each initialization of CA-7.

TRXDV

TRXDV indicates the terminal will be dedicated to transactions scheduled internally by CA-7 functions. At least one such terminal must be defined if either of the following CA-7 functions are used:

- ARF (RESTART, ARF=YES is specified)
- CA-7 XPS SERVER support

BSAM

Indicates that this terminal represents a sequential, print-image browse data set where the master station messages are received. Only one terminal with DEVICE=BSAM may be defined for each initialization of CA-7. This "terminal" should correspond to STANIDS=MASTER on the STATIONS statement.

NAME

Specifies the name which identifies the terminal. The value, up to 7 characters, must match the TNAME value (on the associated LINE statement) that refers to this terminal. NAME is required and has no default. NAME=MASTER should never be specified since this would prevent identification of a master terminal through the STANIDS parameter on the STATIONS statement. (See 5.2.15, "STATIONS Statement" on page 5-40.) See the VTAMID parameter for 8-character VTAM terminal names.

CONS

Designates a terminal from which the /SHUTDOWN command may be entered. CONS is optional and should only be specified for the terminals to be designated as master and alternate master. The device type for master and alternate does not have to be a console. Any 3270x terminal suffices. When omitted, the terminal is assumed not to be a CA-7 alternate master terminal. Value must be one of these:

MASTR

Identifies the CA-7 master terminal. There may be only one master terminal defined for each initialization of CA-7. This should not be confused with STANIDS=MASTER. CONS=MASTR and STANIDS=MASTER rarely refer to the same terminal. See 5.2.15, "STATIONS Statement" on page 5-40.

ALTRN

Identifies an alternate CA-7 master terminal. There may be multiples of these as needed.

Note: This option must be specified on DEVICE=BATCH if you wish to honor SHUTDOWN commands from Batch Terminals (BTI).

LCLSNA

This keyword must be specified as YES when using local SNA terminals under VTAM with any type of 3274-1A controller (for example, 3274-21A and 3274-41A). For all other arrangements, this keyword must be omitted. For virtual terminals, this parameter should not be coded. CA-7 sets the appropriate indicator at connection time. NO is the default. This applies to all SNA terminals, including MASTER.

LINLEN

Specifies the length of output lines for the terminal device. LINLEN is required for non-3270x devices. For 3270x devices, the default value is 80. Value depends on the symbolic device type (designated by DEVICE) and must be one of the following:

- If DEVICE=BATCH, then LINLEN must be line length + 5 (line characters plus ASA control character and record descriptor word).
- If DEVICE=BSAM or DEVICE=3270x, then LINLEN must be 80.
- If DEVICE=CONSL, then LINLEN must be 110.
- If DEVICE=TRLDV, then LINLEN depends on the device type of the CA-7 master station (designated by STANIDS=MASTER on the STATIONS statement).
- If DEVICE=TRLDV or TRXDV, then LINLEN depends on the device type of the CA-7 master station (designated by STANIDS=MASTER on the STATIONS statement).

LOGMODE

Specifies the VTAM log mode name used to identify the kind of information necessary to connect with the terminal. LOGMODE is optional. If omitted, the default is LOGMODE=ZEROS which causes a LOGMODE name of binary zeros to be used. BLANKS causes a name of character blanks to be used. If virtual terminals have been specified and the TERM statement is not coded, LOGMODE defaults to ZEROS.

Note: LOGMODE=ZEROS is required for cross-domain communications in multi-CPU complexes.

MONLIM

Specifies the number of minutes of inactivity allowed to elapse before the terminal is taken out of monitor mode. Monitor mode is used for TIQ and ARTS online functions. This parameter is optional and default is for 2 minutes. Specify zero (0) minutes to disable this feature.

NLINE

Specifies the number of lines per page for the designated device type. NLINE is required for non-3270x devices. For 3270 devices, the default value is 24. Value depends on the device type indicated by DEVICE and must be one of the following:

- If DEVICE=BSAM OR DEVICE=BATCH, then NLINE=60 may be used. This value is variable and may be set at the user's discretion. It should never be less than 10. Leading zeros are not required. The maximum value is 32767.
- If DEVICE=3270x, then NLINE=24 must be used.
- If DEVICE=CONSL, then NLINE=10 may be used. This value is variable and may be set at the user's discretion.
- If DEVICE=TRLDV or TRXDV, then NLINE depends on the device type of the CA-7 master station (designated by STANIDS=MASTER on the STATIONS statement).

PRINTER

Specifies that the device for the terminal being defined is a 3270 associated printer. Value must be PRINTER=YES. PRINTER is optional. If omitted, it is assumed the device is not a printer. If used, DEVICE=3270x must also be specified. For virtual terminals, PRINTER=YES cannot be specified.

RELEASE

Specifies if CA-7 is to release control of the terminal (3270V) to another application under VTAM requesting the terminal. RELEASE is optional and applies to DEVICE=3270V terminals only. Value must be one of the following:

YES

Release the terminal if requested by another application. This is the default.

NO

Do not release the terminal. It is advisable to use RELEASE=NO for VTAM printer devices.

SIMLOG

Specifies if CA-7 is to acquire (issue a SIMLOGON) the terminal under VTAM. SIMLOG is optional and applies to DEVICE=3270V (VTAM) terminals only. Value must be one of the following:

YES

CA-7 is to acquire the terminal through a simulated logon (SIMLOGON) if OPEN=YES was specified on the UCC7VTAM statement. This is the default.

NO

CA-7 does not attempt to acquire the terminal. With this option the terminal operator must initiate the logon.

TIMLIM

Specifies the number of minutes of inactivity allowed to elapse before the terminal is automatically logged off CA-7. In the case of a virtual terminal, the terminal is also returned to VTAM. This parameter is optional and the default is 30 minutes. To disable the logoff feature, specify zero (00) minutes.

Note: When the ISPF interface is being used, the effective TIMLIM for the session will be the greater of 30 minutes or the specified TIMLIM (if a nonzero TIMLIM is coded). Also, there is no indication of a CA-7 time out from an ISPF editor session until the editor session is terminated. This means that a SAVE should be done before the TIMLIM is exceeded or all the edits will be lost.

VLOGOFF

Specifies the program function (PF) or program access (PA) key used to log off CA-7 and return to VTAM. The format for program function keys is PFnn where nn is a number from 01 to 24. The format for program access keys is PAnn where nn is a number from 01 to 03. A two-digit number is required such as PA01; PA1 is invalid. VLOGOFF is optional and is used with DEVICE=3270V only.

VLOGOFF on the GROUP statement may be used to establish a default logoff PF or PA key for all VTAM terminals associated with the group. VLOGOFF on the TERM statement overrides the GROUP definition for that terminal. If not specified in either statement, the terminal operator must issue a /CLOSE command, with no parameters, to return to VTAM.

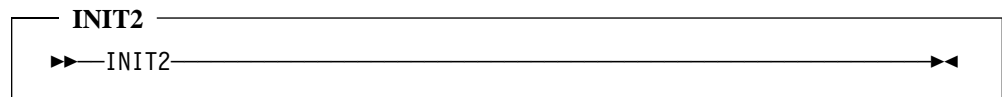
VTAMID

Specifies the VTAM terminal identification name, up to 8 characters. If omitted, VTAMID is assumed to be the NAME value. NAME has a maximum of 7 characters.

5.2.13 INIT2 Statement

The INIT2 statement is optional. When used, it indicates the beginning of the second phase of the initialization process. It is supplied with the skeleton initialization file and requires no changes.

5.2.13.1 Syntax



There are no keywords. INIT2 must begin in column 1.

5.2.14 SECURITY Statement

See the *CA-7 Security Guide* for a complete discussion of the SECURITY statement and its parameters.

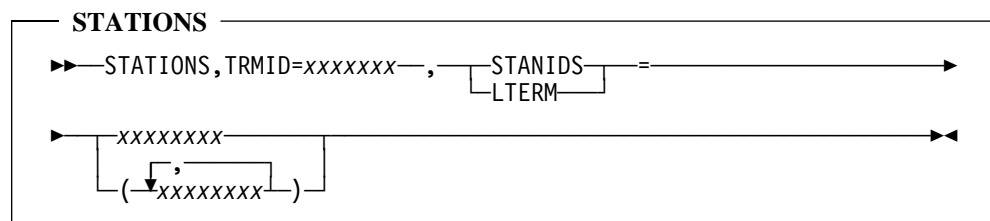
5.2.15 STATIONS Statement

The STATIONS statement defines an installation's workstations or logical terminals, and assigns each to a physical terminal which is already defined by a TERM statement. The STATIONS statement must not be specified for virtual terminals.

Several STATIONS statements are supplied with the skeleton initialization file. Changes may be required to fit the user's terminal network and workstation naming conventions.

A STATIONS statement is always required for each TERM statement (except for virtual terminals), even though the terminal is not used as a workstation. The STATIONS statement may be continued. Keywords are positional and must appear in the order shown.

5.2.15.1 Syntax



Where:

TRMID

Specifies the name of a terminal to which this STATIONS statement is associated. Value must be the terminal name, up to 7 characters, defined by NAME on the TERM statement. TRMID is required and has no default. All TRMIDs must be unique.

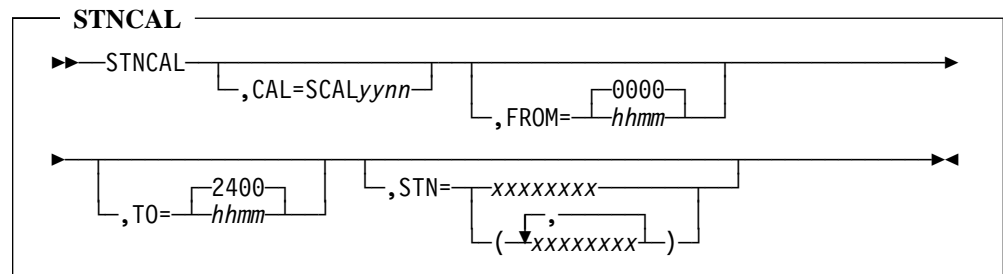
STANIDS | LTERM

Specifies the workstation name, or logical terminal name. Value must be a name, up to 8 characters. Multiple workstation or logical terminal names may be defined as a sublist, enclosed in parentheses and separated by commas. STANIDS is required and has no default. The value given for STANIDS must not be the same as any TRMID value and all STANIDS values must be unique. STANIDS=MASTER defines the CA-7 master station which is to receive the majority of CA-7 messages. The CA-7 master station should not be associated with the OS system master console. There must be one, and only one, STATIONS statement that specifies STANIDS=MASTER. Typically this statement refers to the browse data set or a printer if the browse data set is not used. LTERM can be used interchangeably with STANIDS.

5.2.16 STNCAL Statement

The STNCAL statement defines working hours in a day and working days in a year for the installation's workstations. A separate STNCAL statement can be supplied for each workstation. Alternately, a number of workstations can be combined in one STNCAL statement. A STNCAL statement is optional. If provided, it must immediately follow the last STATIONS statement. If omitted, it is assumed that all workstations are available 24 hours a day for all days in a year and messages to these terminals are queued. When the terminals are not available (that is, not within working days and hours specified), then messages for these terminals go to STANIDS=MASTER.

5.2.16.1 Syntax



Where:

CAL

Specifies the name of a base calendar used for determining working days for the workstation defined by the keyword STN. CAL is optional. If provided, the corresponding calendar name must be defined by a CALBLK statement. If omitted, it is assumed all days are working days for the workstation defined by the keyword STN. Value must be in the format SCALyynn where:

yy

Identifies the year defined for the calendar (00, 01, and so forth). The value must correspond to the value specified for YEAR on the CALENDAR macro statement.

nn

Identifies the unique name assigned by the user at the time the base calendar was generated (that is, the value specified for SCAL on the CALENDAR macro statement).

FROM

Specifies the starting time of a working day. Value must be in the format hhmm where hh are hours of a day (00 to 23) and mm are minutes of an hour (00 to 59). FROM is optional. If provided, it must be less than the value specified for TO. If omitted, the default is FROM=0000.

TO

Specifies the ending time of a working day. Value must be in the format hhmm where hh are hours of a day (00 to 24) and mm are minutes of an hour (00 to 59). TO is optional. If provided, it must be greater than the value specified for FROM. If omitted, the default is TO=2400.

STN

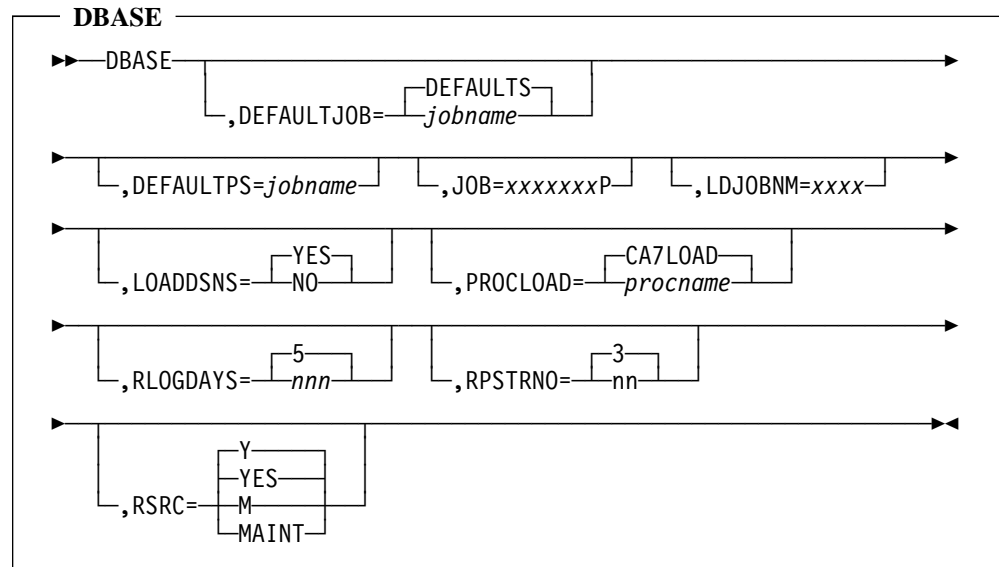
Specifies the workstation name(s) (logical terminals) for which working hours and days are defined. Value must be a name, up to 8 characters. Multiple workstation names may be defined as a sublist, enclosed in parentheses and separated by commas. STN is required and has no default. There must be a corresponding STANIDS value on a STATIONS statement for each of the workstations defined. This must be the last keyword defined in the STNCAL statement.

5.2.17 DBASE Statement

The DBASE statement is required and provides information about database processing options. A DBASE statement is supplied with the skeleton initialization file. It may be modified, if necessary.

Only one DBASE statement can be specified. All keywords are nonpositional and may appear in any order following DBASE.

5.2.17.1 Syntax



Where:

DEFAULTJOB

Specifies a job on the database to be used for default values for all job adds. If a job is defined on the database with this name, the values for fields of this job are used as default values for the new job, with the exception of the MEMBER and UID fields. The default job name is 'DEFAULTS'. See the DEFAULTTPS keyword for the Personal Scheduling exceptions.

DEFAULTTPS

Specifies a job on the database to be used for default values for jobs added through the CA-7 Personal Scheduling facility. If not specified, Personal Scheduling jobs obtain defaults from the regular default job. See the preceding DEFAULTJOB keyword.

JOB

Specifies the name of a CA-7 defined job to be automatically submitted for dumping the primary log data set from disk to tape. Value must be the 8-character job name of the log data set dump job ending with a P. Another job name with the same first 7 characters plus a suffix of S is used for dumping the secondary log data set. Both jobs must be defined in the CA-7 database and the JCL must reside in one of the JCL data sets defined to CA-7 in an initialization file JCL statement. JOB is optional. If omitted, automatic submission of jobs for execution is suspended when the log data set becomes full and CA-7 stops. For additional information, see 8.1, “Log Data Set Management” on page 8-2.

LDJOBNM

Specifies the first 4 characters to be used for load-only job names generated for top line LOAD commands. Four characters must be specified. If used, the load-only job name becomes xxxnnnn, where xxxx is the LDJOBNM value and nnnn is the CA-7 assigned job number. All load-only jobs then execute with this name. LDJOBNM is optional. If omitted, the normal job name is used.

Note: When this keyword is used, existing JOB statements for jobs being loaded must allow eight positions for job name; otherwise JCL syntax errors result.

LOADDSNS

Specifies whether the CA-7 LOAD process should build data set information and requirement entries for jobs. This only applies to jobs that are marked as MAINT=Y on the DB.1 screen.

YES

Indicates the LOAD process adds DD information to the CA-7 database for the MAINT=Y jobs. This is the default.

NO

Indicates the LOAD process ignores any DD information for MAINT=Y jobs; bypassing adding the information to the CA-7 database.

If this option is specified as NO, consider the following:

Benefits:

- Smaller CA-7 database files.
- Faster LOAD processing.
- No need to mark data sets PERM, since MAINT=Y is used.
- Reduced maintenance of the database files.
- Data set triggers can be used if the data sets are added to the database manually or the job is MAINT=N.

Functions not available for MAINT=Y jobs when LOADDNS=NO is used:

- Automatic building of data set requirements during the CA-7 LOAD process. LJOB,JOB=xxx,LIST=RQMT show no data set requirements unless they are added manually.
- Data set information added to the database files (IDS and SASDS). The LDSN does not reflect any data sets being used by the jobs.
- Job data set DD/dsn information. LJOB,JOB=xxx,LIST=STEPDD show only the steps for the job, not data set information.
- SASSBK00 XREF reports for data set/job.
- Forecast commands for resources do not reflect any information based on data sets (tape pull lists, and so on). FTAPE (and FQTAPE) are no longer useful.
- Workload balancing and workload planning no longer have tape information to work with (the TAPE1 and TAPE2 values on the DB.1 (JOB) panel always show 0). You are unable to use WLB to balance these jobs based on tape drive usage (unless the tape values are filled in manually).

PROCLOAD

Specifies the 1- to 8-character procedure name to be inserted to cause CA-7 Load processing. The Load processing builds job profiles from submitted JCL. The default is CA7LOAD which is generated as part of the CA-7 installation.

RLOGDAYS

This parameter specifies the number of days of CA-7 run log data that are kept for access online. This statement is optional and has a default of 5. The maximum value is 365. Here are some considerations when increasing this value:

- **Caution** *Each day can use up to 30 tracks (sometimes more) of trailer queue space, possibly requiring a larger trailer queue allocation.*
- The LRLOG command may take longer as the number of days is increased (because there is more data to be processed).

RSRC

Specifies the type of Virtual Resource Management (VRM) support that is activated during CA-7 initialization.

YES

Full VRM support. This is the default.

MAINT

Allows maintenance of the VRM database without resource management during the job submission process.

5.2.18 ALOG1 Statement

The ALOG1 statement specifies the name of the primary log data set. It is supplied with the skeleton initialization file, but may require changes by the user.

The ALOG1 statement is always required unless the initialization is for a maintenance or batch only run (MAIN or REPT was specified for RUNOPT on the INIT statement). Continuation of the ALOG1 statement should never be necessary.

5.2.18.1 Syntax

<div style="display: flex; align-items: center;"> <div style="margin-right: 20px;"> ALOG1 </div> <div style="flex-grow: 1; border-bottom: 1px solid black; position: relative;"> <div style="position: absolute; left: 0; top: -5px;">▶▶</div> <div style="position: absolute; right: 0; top: -5px;">◀◀</div> </div> </div> <div style="margin-top: 5px;"> ▶▶ ALOG1,DSN=x...x◀◀ </div>

Where:

DSN

Specifies the fully qualified data set name of the primary log data set. The value, up to 44 characters, must correspond to the DSN given in the UCC7LOG DD statement in the CA-7 JCL. DSN is required and has no default.

5.2.19 ALOG2 Statement

The ALOG2 statement identifies the alternate (secondary) log data set. It has the same format as the ALOG1 statement except for the statement keyword identifier which is ALOG2 for this statement. It is supplied with the skeleton initialization file but may require change or removal by the user.

- If a secondary log data set is defined, then both primary and secondary logs must be disk data sets and they must reside on the same volume with unique data set names.

Note: An additional DD is not required in the CA-7 JCL for this secondary log data set. The DSN value specified is used, in combination with the ddname UCC7LOG, to access the data set. That ddname is used to define the primary log data set (ALOG1) in the CA-7 execution JCL.
- If the primary log data set is a tape data set, then no secondary log is necessary and the ALOG2 statement must be omitted. Primary log on tape is not recommended.
- See 6.2.2, “Log Usage” on page 6-16 for further discussion of how these log data sets are offloaded.

5.2.20 RESTART Statement

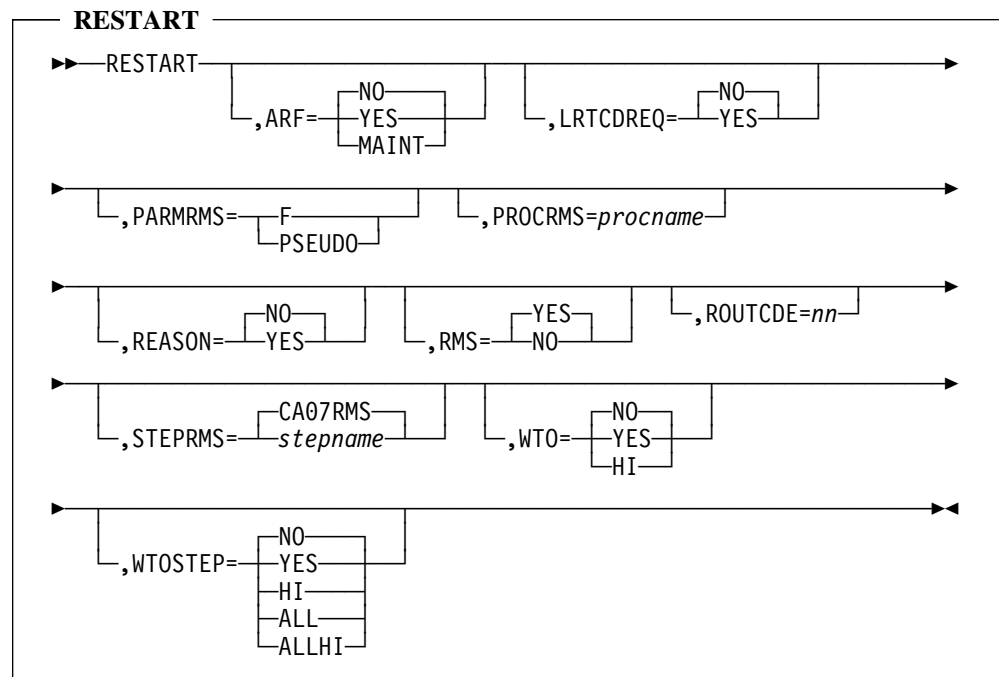
The RESTART statement identifies the available option for job restart processing.

The RESTART statement is optional; however, if the CA-11 interface is desired, the RESTART statement must be in the initialization file.

See *CA-7 Interfaces Guide* for additional information about using the CA-11 interface.

Obsolete keywords USAGE and AUTOOF are now set from the CA-11 Options Table. If these keywords are coded in the RESTART statement, they are ignored.

5.2.20.1 Syntax



Where:

ARF

Specifies whether the Automated Recovery Facility is available.

NO

Indicates that ARF is not available. This is the default.

YES

Indicates that ARF is available. If this value is specified, ARF monitors and responds to job events.

MAINT

Indicates that ARF is available for data base maintenance only. If this option is specified, ARF database definitions may be accessed, but ARF does not monitor or respond to job events.

LRTCDREQ

Specifies whether the LRTCD field on the XRST screen is required.

NO

The LRTCD field is not required. This is the default.

YES

The LRTCD field is required. You can specify a value of 1 for the stepname, and this causes the LRTCD field to be ignored. This would be necessary if restarting in the first step of the job.

PARMRMS

Specifies the parm information to be passed to the RMS step at job submission. Using 'F' passes a PARM='F' to the RMS step. This is highly discouraged since it forces CA-11 to do format processing for every job that is submitted. This causes extreme overhead in CA-11 execution and fills up the CMT much quicker (requiring a compress of the CMT PDS).

Using 'PSEUDO' passes a PARM='P,PSEUDO=YES' to the RMS step, unless restart (for example, XRST) overrides this by specifying an F in the SET PARM field.

PARMRMS has no default. If it is not specified, then the PARM is set to a 'P' for initial runs. For restarts, the PARM is set according to the XRST and CA-11 options.

PROCRMS

Specifies the 1- to 8-character procedure name to be inserted for jobs flagged for INSERT-RMS on the DB.1 screen. If CA-11 is not installed, insertion of the RMS step does not occur even though the DB.1 screen field requests it. The default procedure name is taken from the CA-11 Options Table, which should have been created during CA-11 installation. Regardless of the name used, the symbolic parameter &TYPRUN must exist in the procedure referenced by PROCRMS.

REASON

Specifies whether a reason code must be input when doing a job restart. The reason code entered, up to 40 characters, is written to the CA-7 run log and, if CA-11 is available, passed on to the CMT. The value must be YES or NO. The default is NO.

RMS

Specifies whether CA-11 is used for restarts if CA-11 is available. If RMS=NO is specified, the RMS step is not inserted and restart commands/screen do not display or post CA-11 data. However, the ARTS command may be used. Value must be YES or NO. The default is YES, which causes CA-11 to be used, if CA-11 is present.

ROUTCDE

Specifies route code for the job and step termination WTOs if the WTO= and/or WTOSTEP= keywords have been specified (WTOs CA-7.SMF3 and CA-7.SMF4). The value must be a number between 1 and 16. The default is ROUTCDE=1 which sends the WTOs to the master console.

STEPRMS

Specifies a 1-to 8-character step name for the CA-11 RMS step. If a step name is not specified, the default stepname is CA07RMS.

WTO

Specifies whether a WTO is to be issued when a CA-7 submitted job terminates unsuccessfully. See the *CA-7 Message Guide* for message CA-7.SMF3.

NO

Indicates that CA-7 does not issue a WTO at unsuccessful job completions. This is the default.

YES

Indicates that CA-7 issues a WTO.

HI

Indicates that CA-7 issues a highlighted and nonscrollable WTO.

WTOSTEP

Specifies whether a WTO is to be issued for job steps that CA-7 tracks (see message CA-7.SMF4). The unsuccessful steps are those that terminate with an abend, JCL error, or condition code that fails the job=level code form the DB.1 (JOB) screen. Steps that fail the #SCC tests are not marked as unsuccessful (though they cause a job WTO to be issued if that option has been specified). LOAD only jobs produce a WTO only if the LOAD step fails with something other than its usual JCLERR.

NO

Indicates that a step fail WTO is not issued. This is the default.

YES

Indicates issue a WTO for bad step terminations.

HI

Indicates issue a highlighted and nonscrollable WTO for bad step terminations.

ALL

Indicates issue a WTO for every step completion for every CA-7 tracked job regardless of the termination condition except for LOAD only jobs.

ALLHI

Indicates issue a WTO for every step completion for every CA-7 tracked job regardless of the termination condition except for LOAD only jobs. Also, WTOs are highlighted and nonscrollable.

Note: If ALL or ALLHI is used, a large number of WTOs are issued depending on the number of jobs being tracked by CA-7 and the number of steps in each job.

5.2.21 SCHEDULE Statement

The SCHEDULE statement establishes values for the parameters which govern the automatic scheduling and submission control functions of CA-7. This feature is known as schedule scan. The initial values define:

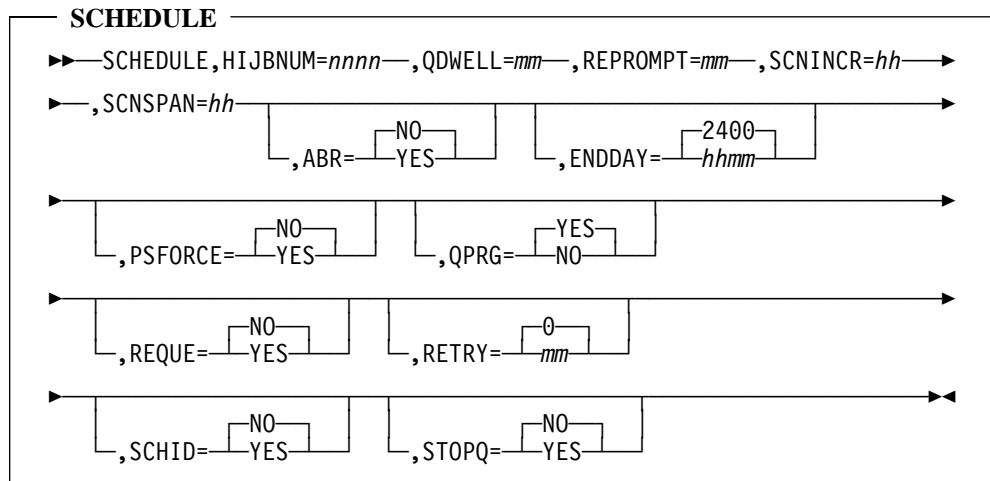
- The maximum job number to be assigned to a CA-7 submitted job.
- The schedule scan wake-up interval and time span to be searched for work to schedule.
- The time interval to elapse between reprompts.

The SCHEDULE statement is supplied with the skeleton initialization file. The specified values may be changed at the user's discretion.

The SCHEDULE statement is always required when CA-7 is initialized for the purpose of automatically scheduling and controlling production. If the initialization is for a maintenance or batch-only run (RUNOPT=MAIN or RUNOPT=REPT was specified on the INIT statement), then the SCHEDULE statement may be omitted. The SCHEDULE statement may be continued.

All keywords are nonpositional and may appear in any order following SCHEDULE.

5.2.21.1 Syntax



Where:

HIJBNUM

CA-7 assigns its own job numbers to jobs in a manner similar to the way JES assigns job numbers. This parameter specifies the highest allowable CA-7 assigned job number. HIJBNUM is required and has no default. The maximum value is 9999.

Note: When this value is changed, a COLD type of start of CA-7 is required.

QDWELL

Specifies a safety factor for job scheduling which ensures that a job is not in a late status when it is initially entered into the queue. Value must be given in minutes (0 to 60). The recommended value is 30 minutes. QDWELL is required and has no default.

QDWELL becomes important when SCNSPAN and SCNINCR specify the same interval. For example, if both were specified as two hours but QDWELL was not specified and CA-7 was initialized at 0800, schedule scan would wake up at 08:00, 10:00, 12:00 and so forth. Work scheduled for processing within the next two hours would be loaded into the queues each time. Work scheduled to start at 10:01, 12:01, or 14:01 would not be loaded until 10:00, 12:00 and 14:00 respectively. This could cause work to be flagged as late because of the small amount of time between arrival in the queue and when the work is scheduled to start (only one minute in the preceding example).

REPROMPT

Specifies the time interval between wake-ups for reprompting for late work. Value must be given in minutes (0 to 60). The recommended value is between 15 to 30 minutes, depending on the installation's needs. A value of 0 turns off reprompting. REPROMPT is required and has no default.

SCNINCR

Specifies the time interval for schedule scan wake-up. Value must be given in hours (1 to 24), and must be less than or equal to the SCNSPAN value. SCNINCR is required and has no default.

Note: See note under SCNSPAN below.

SCNSPAN

Specifies the span of time beyond the time-of-day of each schedule scan wake-up to be searched for jobs to be scheduled. Any job with a deadline start time which occurs within that span is scheduled into the queues. Value must be given in hours (1 to 24), and must be equal to or greater than the SCNINCR value. SCNSPAN is required and has no default.

Note: The example in the following figure shows schedule scan becoming active every two hours. When schedule scan becomes active, all work scheduled to start within the next four and one-half hours (SCNSPAN plus QDWELL) is loaded into the queues. The extra half-hour scheduling is caused by the QDWELL keyword specifying an overlap of 30 minutes.

ABR

If YES is coded, the format of initial requirements scan output and prompt messages is abbreviated. The default is NO.

ENDDAY

Specifies the end of a processing day (for use in automated performance analysis statistical reporting). Value must be given as hours and minutes (hhmm) with 2400 as maximum. ENDDAY is optional and defaults to 2400.

PSFORCE

Indicates CA-7/Personal Scheduling job completion status.

NO

Indicates that CA-7/PS submitted jobs are not forced complete automatically. This is the default.

YES

Indicates any jobs submitted using the CA-7/Personal Scheduling SUBMIT function are always considered successfully completed, regardless of actual completion of the job on the CPU.

QPRG

Specifies whether a job in the ready queue is automatically requeued to the request queue if an SMF type 26 (Job Purge) record is received for that job. When YES is specified or defaulted, the job is requeued and shows a status of R-JCLERR.

Note: While most JCLERR conditions can be automatically requeued by using QPRG=YES, there are some that cannot be and require a manual requeue. This is because of the lack of information in the IBM SMF purge record. For example, DSIP=SHR can be caught, but DISP=SRH cannot. Another situation that can leave a job in the ready queue with a R-JCLERR status is the use of the \$RXEQ JES command.

A reload of the JES spool can cause a purge record to be produced for the jobs reloaded and for jobs in the CA-7 ready queue. A JCLERR indication will be noticed, and if using QPRG=YES, the job will be requeued back to the request queue with JCLERR status.

REQUE

Specifies whether a job that is abended with a S222 and has a zero CPU time is left in the ready queue. If a requeuing package performs an operator requeue in JES, then specifying YES causes CA-7 to leave the job in the ready queue awaiting the job start record. The default is NO.

RETRY

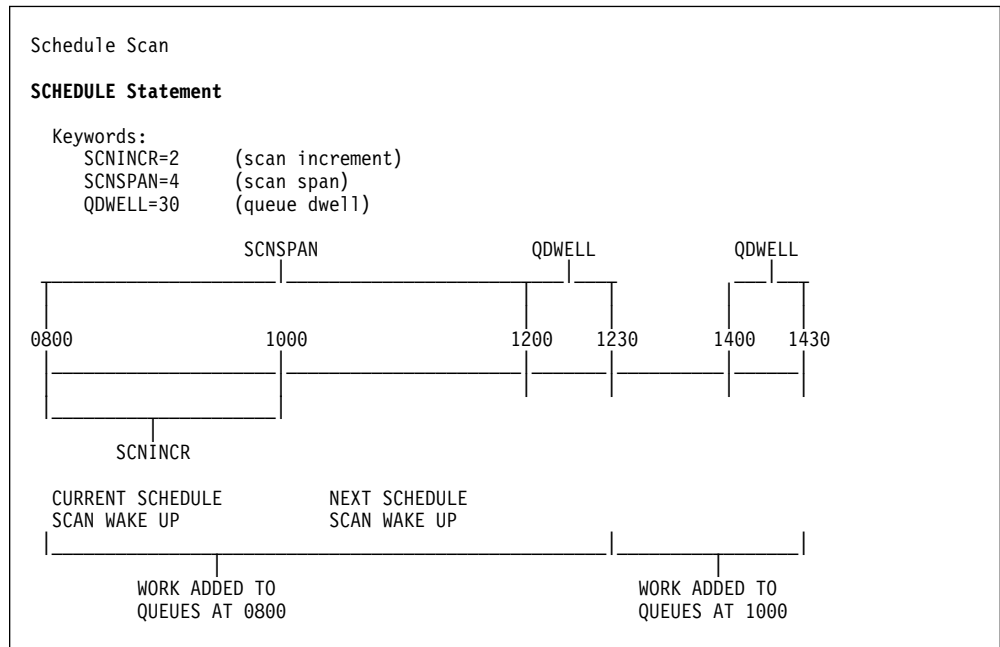
Specifies the time interval between wake-ups for reattempting to attach JCL and requirements for jobs in RETRY status. Jobs may be left in RETRY status if there was a dynamic allocation (SVC 99) failure trying to access the JCL library when attempting to attach the JCL for the job. A value of 0 turns off the RETRY cycle. The RETRY parameter is optional and defaults to 0. Value must be given in minutes (0 to 60).

SCHID

Specifies whether a SCHID= parameter is required for DEMAND(H), RUN(H), and LOAD(H) commands. The default is NO.

STOPQ

If YES is coded, normal job movement in the request and ready queues is suspended. This has the same effect as the top line command STOP,Q=ALL. To resume normal job flow, use the START,Q=ALL command.



5.2.22 CPU Statement

The CPU statement is used to define the ddname, in the CA-7 execution JCL, to which CA-7 writes the JCL for jobs to be submitted.

- For nonshared spool multi-CPU systems, the CPU statement establishes the relationship between the submit data set and the ICOM which submits the work. In this situation, a CPU statement for each submit data set is required for each CPU that is to process CA-7 jobs.
- For single CPU systems, or multi-CPU systems with shared spool, throughput of submit processing can be improved by using a JES internal reader. With this approach, CA-7 submits the JCL directly to the internal reader instead of the submit data set (that is, ICOM does not submit the jobs). In this case, only one CPU statement may be required for shared spool systems.

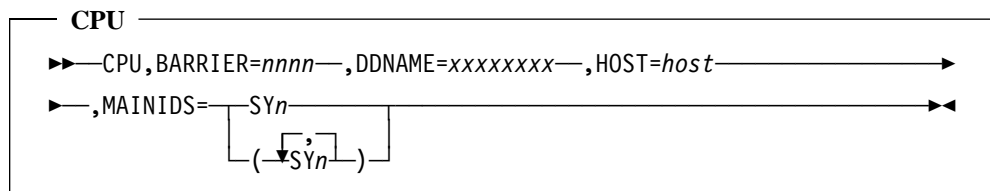
For example:

An operating environment may consist of two shared spool CPUs and one nonshared spool CPU with CA-7 running on one of the shared spool CPUs. In this case one CPU statement is required for the two shared CPUs with a ddname corresponding to an internal reader. Another CPU statement is needed for the nonshared spool system with a ddname corresponding to a submit data set. Also, see the PARM for ICOM's execution in 6.6.2, "ICOM PARM Values" on page 6-27.

A CPU statement is supplied with the skeleton initialization file. This statement must be modified as necessary to fit the user's environment.

The CPU statement may be continued. All keywords are nonpositional and may appear in any order following CPU.

5.2.22.1 Syntax



Where:

BARRIER

Specifies the maximum number of CA-7 submitted jobs to be allowed in the host system job queue at any one time. Value must be a decimal number, up to 4 digits. BARRIER is required and has no default.

DDNAME

Specifies the name of the DD statement in the CA-7 JCL which defines this submit data set or internal reader. Value must be the 8-character ddname to which job submission for this CPU is done. DDNAME is required and has no default.

Note: A DDNAME=UCC7IRDx denotes an internal reader for the host job entry subsystem, where x can be any valid character. A corresponding JCL statement must be added to the CA-7 JCL with the DDNAME pointing to an internal reader.

HOST

Specifies the job entry subsystem to which JCL is submitted. Value must be JES2 or JES3. HOST is required and has no default.

MAINIDS

Specifies the CA-7 main ID which is assigned to the submit data set or internal reader. Multiple main IDs may be specified as a sublist, enclosed in parentheses and separated by commas. MAINIDS is required and has no default. Value must be in the format SYn where n may be:

1-7

A main ID number used on the DB.1 screen for each job to indicate the CPU(s) on which a job may or may not execute. Specific main IDs, SY1 through SY7, may appear in multiple CPU statements. This causes the JCL for a job with a matching ID to be written to the submit data set or internal reader specified by the DDNAME keyword.

0

Indicates where the JCL for jobs without a specific MAINID (1 to 7) or with MAINID=ALL on the DB.1 screen is written.

Indicates a default. Those jobs with main IDs defined on the DB.1 screen that do not match any other CPU statement MAINID are written to this default submit data set or internal reader. If multiple CPU statements are used, then SY* should only be specified on one of them.

5.2.23 CALENDAR Statement

The CALENDAR statement is used to identify the optional CA-7 calendar PDS where copies of CA-7 base calendars are stored. This data set is required only if you wish to have online access to base calendars using the DB.2.8 Calendar Maintenance screen, or through the Calendar facilities available to Unicenter TNG.

See member PDSCAL in the CA-7 Sample JCL Library for model JCL to allocate a calendar PDS. Use the name of the allocated library in the CALENDAR initialization file statement. When you start CA-7 with the modified initialization file, all CA-7 base calendars defined with CALBLK statements are copied to the calendar PDS in a new format. These existing calendar load modules will NOT be altered or deleted. New base calendars created through online facilities do not require CALBLK statements.

5.2.23.1 Syntax

CALENDAR
►► CALENDAR, DSN=x...x ◀◀

Where:

DSN

Specifies the fully qualified name of the CA-7 Calendar PDS (partitioned data set).

5.2.24 JCL Statement

The JCL statement provides information about data sets containing the execution JCL for jobs to be submitted by CA-7. JCL statements are supplied with the skeleton initialization file. These statements may be modified as needed to fit the user's environment.

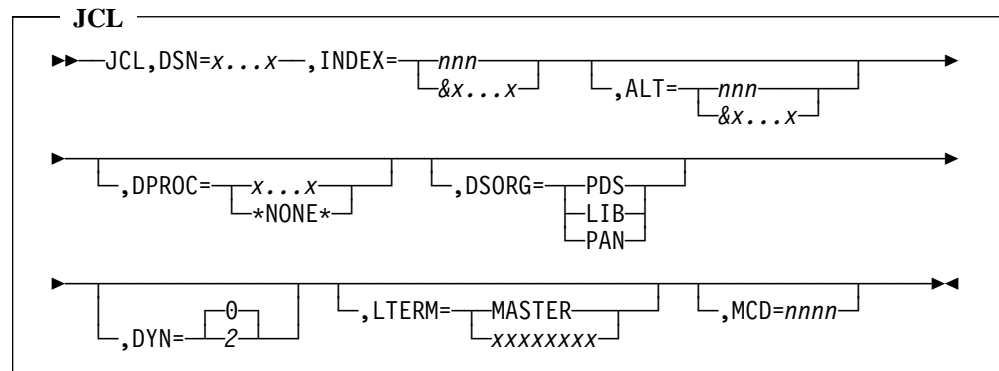
JCL statements may be specified for every initialization of CA-7. Multiple JCL statements may be specified if execution JCL resides in more than one data set.

JCL statements are stored in the CA-7 database. If JCLDEFS=INITFILE was specified on the RESIDENT statement, JCL statements specified in the initialization file are used. If JCLDEFS=VSAM was specified on the RESIDENT statement, JCL statements stored in the CA-7 database are used and JCL statements specified in the initialization file are ignored.

The JCL statement may be continued.

All keywords are nonpositional and may appear in any order following JCL.

5.2.24.1 Syntax



Where:

DSN

Specifies the fully qualified name of a data set containing execution JCL to be submitted by CA-7. DSN is required and has no default.

INDEX

Specifies the unique identifier to be assigned to the JCL data set. INDEX is used to equate a number or a symbolic value with a data set containing execution JCL. INDEX is required.

nnn

An index number is referred to as a JCLID on the DB.1 screen and in certain commands. This number may range from 0 to 255. Leading zeros are required only for CA-Librarian data sets. INDEX number 254 is reserved for the override library, and 255 is reserved for the HELP data set. The default JCLID on the DB.1 screen is 0. It is recommended that one JCL data set be given an INDEX value of 0 so that the DB.1 screen JCLID field does not have to be entered for jobs with JCL residing in that data set.

&x...x

A symbolic index is referred to as a JCLLIB on the DB.1 screen and in certain commands. A symbolic index consists of an ampersand (&) followed by up to 15 alphanumeric characters. If symbolic indexes are used, JCL libraries must be dynamically allocated and DYN=2 must be specified. Symbolic indexes can only be used with PDS data sets. Symbolic value &HELP is reserved for the HELP data set and may be specified instead of index number 255. Attributes of a JCL data set referenced by a symbolic index may be changed with the /JCL command. For more information, see /JCL in the *CA-7 Commands Guide*.

ALT

Specifies the INDEX value from a previously defined JCL library that will be searched prior to this one. This works exactly like DD statement concatenation where the ALT is first in the sequence, but is supported for only one level. Since validation of this parameter is done while the internal table of JCL libraries is being built, the alternate library must be defined in the initialization file prior to the statement which references it as an alternate. ALT and INDEX values cannot be equal in any one JCL statement. Refer to the Alternate JCL Libraries section of the *CA-7 Database Maintenance Guide* for a discussion of how this option may be used. ALT is optional.

DPROC

Specifies the data set name of a CA-Driver procedure library to be concatenated above the libraries allocated with the CARPROC DD statement whenever CA-Driver is invoked for JCL in the library defined.

DPROC=*NONE* indicates that CA-Driver is not to be invoked for JCL in the library defined.

DSORG

Specifies the organization of the JCL data set being defined. DSORG is optional. If omitted, the default is DSORG=PDS. Value, if used, must be one of the following:

PDS

Indicates the data set is an OS partitioned data set.

LIB

Indicates the data set is a CA-Librarian library. A special DD statement is required for such libraries using the name JCLnnn, where nnn is three digits (leading zeros required) matching the value for INDEX. See CA-Librarian in the *CA-7 Interfaces Guide*.

PAN

Indicates the data set is a CA-Panvalet library. A special DD statement is required for such libraries using the name JCLnnn, where nnn is three digits (leading zeros required) matching the value for INDEX. See CA-Panvalet in the *CA-7 Interfaces Guide*.

DYN

If dynamic allocation of JCL data sets is used (see the PERFORM keyword under 5.2.4, “INIT Statement” on page 5-13), this parameter may be used to specify how the JCL data set defined in this statement is allocated. This option is only valid for PDS JCL data sets (not valid for CA-Panvalet or CA-Librarian files). Value, if used, must be one of the following:

0

This is the default and indicates the JCL data set is allocated as DISP=SHR for read-only access and as DISP=OLD for update access.

2

This indicates that the JCL data set is allocated as DISP=SHR for update and for read access. However, for update, ENQs are issued in an attempt to serialize updates with TSO. This allows the data set to be browsed with TSO while CA-7 updates it, and allows TSO updates on one member and CA-7 updates on another. If a program other than TSO updates a data set while CA-7 is allocating it with DISP=SHR, simultaneous updates are possible if the other program does not respect the ENQ conventions used by CA-7. The ENQs use SYSDSN and SPFEDIT as major names.

DYN=2 can only be used if the dynamic allocation option is used. This option must be used for libraries defined with symbolic indexes.

Note: CA-Panvalet and CA-Librarian files are read-only files through CA-7. PDS files can be read and updated by CA-7. For CA-Panvalet and CA-Librarian data sets the interface must be installed for CA-7 to access the JCL. See CA-Librarian and CA-Panvalet in the *CA-7 Interfaces Guide*. The DASD on which PDS files reside must be made accessible to CA-7 by including a U7volser DD statement in the CA-7 JCL, unless the dynamic allocation feature is used. See the PERFORM keyword under 5.2.4, “INIT Statement” on page 5-13. CA-Panvalet and CA-Librarian data sets require DD statements in the CA-7 execution JCL which include the appropriate DSN value. See 6.1, “CA-7 Execution” on page 6-2 for more information.

LTERM

Specifies the logical terminal to which prompt messages should be routed for jobs that are submitted using JCL from this JCL data set.

- The DB.1 screen LTERM field can be used to define a logical terminal for a specific job.
- LTERM is optional. If omitted and not defined on the DB.1 screen, the default is LTERM=MASTER, the logical terminal that receives the majority of CA-7 messages.
- If a logical terminal other than MASTER is specified, it must match a STANIDS value on one of the STATIONS statements.

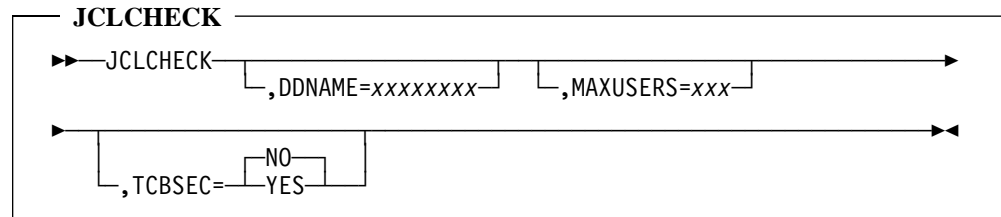
MCD

Specifies the management code used to access PROD2 members of CA-Librarian master files. Valid only if DSORG=LIB and cannot be used if the library is defined using a symbolic index. If specified, the value must be 4 characters in length (leading zeros required) and cannot exceed 8768. See the CA-Librarian documentation for further details.

5.2.25 JCLCHECK Statement

The JCLCHECK statement is required if the CA-7 CA-JCLCheck interface is to be used.

5.2.25.1 Syntax



Where:

DDNAME

Specifies the DDNAME for a DD statement in the execution JCL for CA-7. This statement should point to a sequential data set where installation specific runtime options for CA-JCLCheck reside. The data set should be defined:

```
DCB=(LRECL=80,RECFM=FB,BLKSIZE=80 * nnnn)
```

Only the following user-specified options are allowed:

COBEXIT, EASYRDR, ASM2, DESTCHK(x), TMS, TLMS, SEV(xx), INDEX(xx,xx,...), REVINDEX(xx,xx,...), PX, PX(NOR), NOPX, HCD, VSAM

In addition, the AUTOPROC option is allowed if the following conditions are true:

1. A SYSPROC DD statement is not coded in the CA-7 startup procedure.
2. CA-7 is operating in a JES2 environment.

Note: If there is no SYSPROC DD statement pointing to procedure libraries and the AUTOPROC option is not used, then an error message is produced and the interface is disabled.

For further information on CA-JCLCheck options, see the CA-JCLCheck documentation.

MAXUSERS

Specifies a maximum limit on the number of concurrent users of the interface. If this value is not coded, then there is no limit on the number of concurrent interface users other than whatever practical limits may obtain due to resource constraints.

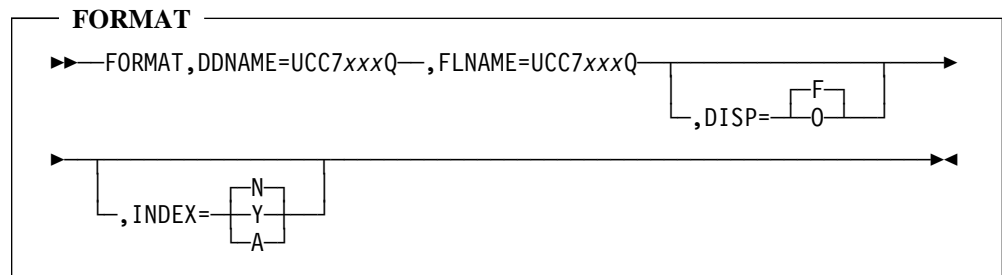
TCBSEC

This option enables the TCB security option. The default is NO. CA-7 must have at least EXTERNAL=LOGON coded in the SECURITY statement for this to work. With this option enabled, TCBSEC=YES, any invocation of the CA-JCLCheck interface puts the security token of the user who issued the command in the TCBSERV field for the current TCB. This allows the external security package access to the security token for validation during CA-JCLCheck processing. Check the documentation for your external security package for information on how to enable checking from the TCBSERV field.

5.2.26 FORMAT Statement

The FORMAT statements identify the CA-7 processing queues. They may also specify formatting options to be applied at initialization time. The FORMAT statements are supplied with the skeleton initialization file and should require few changes. Nine FORMAT statements are used to define the queues. All nine are required for every online initialization of CA-7. No continuation should be necessary. All keywords are nonpositional and may appear in any order following FORMAT.

5.2.26.1 Syntax



Where:

DDNAME

Indicates the DD statement which defines each queue in the CA-7 JCL. DDNAME is required and has no default. The first four characters must be UCC7 and the last character must be a Q. Value must be one of the following (one per FORMAT statement):

UCC7ACTQ

Active queue

UCC7DQTQ

Disk queue table

UCC7PREQ

Preprocess queue

UCC7PRNQ

Prior-run queue

UCC7PSTQ

Postprocess queue

UCC7RDYQ

Ready queue

UCC7REQQ

Request queue

UCC7SCRQ

Scratch queue

UCC7TRLQ

Trailer queue

FLNAME

Specifies the internal name used by CA-7 to reference each queue. On the FORMAT statement for the scratch queue (DDNAME=UCC7SCRQ), FLNAME must be omitted. Otherwise, FLNAME is required and has no default. The first four characters must be UCC7 and last character must be a Q. Value must be one of the following (one per FORMAT statement) and must appear with the associated DDNAME value:

UCC7ACTQ

Active queue

UCC7DQTQ

Disk queue table

UCC7PREQ

Preprocess queue

UCC7PRNQ

Prior-run queue

UCC7PSTQ

Postprocess queue

UCC7RDYQ

Ready queue

UCC7REQQ

Request queue

UCC7TRLQ

Trailer queue

DISP

Specifies the formatting option to be used during initialization for a TYPE=COLD start. See 6.3.1.3, “COLD Start (Clear Workload Restart)” on page 6-20. Value must be one of the following:

F

Causes all tracks in the specified queue to be formatted and cleared.

O

Causes formatting to be suppressed. This is an alpha character, not a zero.

A TYPE=FORM start overrides this option and forces formatting of all queues. For a TYPE=COLD start, all data is deleted for the queue even though it is not formatted (less I/O required). However, a DISP=O specifically for the PRNQ causes its data to be kept. This can be useful to keep the PRNQ data even on a TYPE=COLD start.

DISP is optional. Its use depends on the type of initialization being performed (TYPE parameter on the INIT statement or the EXEC statement in the CA-7 JCL). If TYPE=COLD, either option may be used and the value specified is applied by the initialization process. (Use of DISP=O when the queues have already been formatted reduces the initialization time.) DISP may be omitted if TYPE=WARM since it is ignored by the initialization process.

Some performance benefit may be gained if the scratch queue and the disk queue table (DQT) queue are defined as temporary data sets on VIO devices. If these queues are defined in this way, the corresponding FORMAT statements should indicate DISP=F so that these queues will be formatted on an ERST start.

It is recommended that DISP=O be used at all times for queues other than the scratch and DQT queues unless otherwise directed by a CA-7 technical consultant. However, if TYPE=FORM, all queues are reformatted regardless of this specification.

INDEX

Indicates whether none of the queue, the entire queue or only an index is to be built in memory for the specified queue. Using INDEX=Y or A improves the response time for queue requests, causing fewer queue lockouts. INDEX is valid only for the status queues (PRE, REQ, RDY, ACT, POST, and PRRN).

The trailer queue, though not a status queue, will always be kept in memory when CA-7 is executing on OS/390, MVS/XA, or MVS/ESA systems. The entire queue is not loaded, but the individual data is loaded in memory as it is referenced. This enables CA-7 to keep the used portion of the trailer queue in memory without having to reserve storage for the size of the entire queue. The memory-resident trailer queue is placed in storage above 16M and will usually use 1M - 3M of storage. An INDEX option is not used for the trailer queue.

N

Causes no index or memory-resident queue to be built. This is the default.

Y

Causes an index to be built in memory for the specified queue. This option is recommended only for the request, ready and active queues and then only if INDEX=A is not used. INDEX=A is a much better performance option. It may also be used for the preprocess, postprocess, and prior-run queues, but is ignored for the remaining queues. INDEX=Y is optional.

A

Indicates that the entire queue is to be memory-resident. When used, all input I/O requests are serviced only from the memory-resident queue. All output I/Os are performed against both the memory-resident and the DASD-resident queues for restartability purposes. It is recommended that INDEX=A be used for the request, ready, and active queues. The trailer queue data is kept in storage in memory on OS/390, XA, and ESA operating systems. This storage is above 16M.

5.2.26.2 Memory Requirements

INDEX=Y for a large queue may require an increase in region size and in the CWORK parameter on the EXEC statement in the CA-7 JCL. The equation for determining memory requirements (C) for an index is:

$$C = (T * B * 70) + 36$$

Where:

T = Tracks allocated for the queue

B = Records per track from the DAIO statement, RECORDS keyword

For example, a 10-track queue on a 3390 would require:

$$(10 * 33 * 70) + 36 = 23136 \text{ bytes}$$

INDEX=A for a large queue may require an increase in region size and in the CWORK parameter on the EXEC statement in the CA-7 JCL. The equation for determining memory requirements (C) for a memory-resident queue is:

$$C = (T * B * 1022) + 36$$

Where:

T = Tracks allocated for the queue

B = Records per track from the DAIO statement, RECORDS keyword

For example, a 10-track queue on a 3390 would require:

$$(10 * 33 * 1022) + 36 = 337296 \text{ bytes}$$

See 10.2.6, “Memory-Resident Queues” on page 10-7.

Note: Before implementing memory-resident queues or indexes, make sure the queue data set for which the option is to be used is not overallocated. Excess in DASD allocations unnecessarily inflates the memory requirements. Memory is allocated for every track allocated. See the /DISPLAY,Q=ALL command and the CA-7 Queue Allocation Usage Profile report.

5.2.27 DAIO Statement

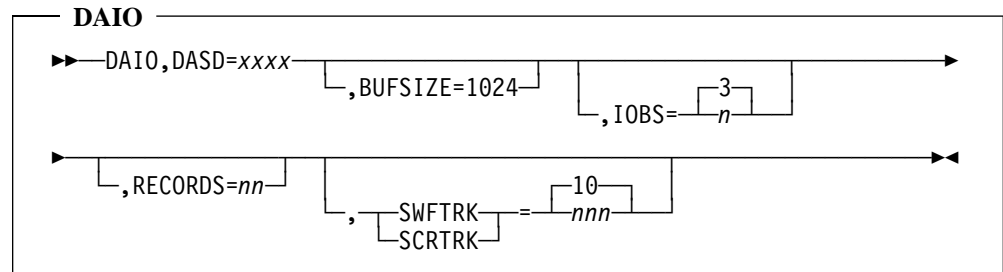
The DAIO statement defines attributes, resource requirements and type of DASD being used for the queue data sets.

The DAIO statement is supplied with the skeleton initialization file. Only the DASD device type specification and records per track should be changed initially, if necessary.

The DAIO statement is required for every initialization of CA-7. It must not appear more than once in the initialization file. No continuation should be necessary.

All keywords are nonpositional and may appear in any order following DAIO.

5.2.27.1 Syntax



Where:

DASD

Specifies the type of direct access device on which the queue data sets reside. DASD is required and has no default. Value must be one of the following:

3330

Indicates a 3330-I or 3330-II DASD device

3350

Indicates a 3350 DASD device

3375

Indicates a 3375 DASD device

3380

Indicates a 3380 DASD device

3390

Indicates a 3390 DASD device

9345

Indicates a 9345 DASD device

BUFSIZE

Specifies the physical block size for the queue data sets. Value must be 1024 bytes (BUFSIZE=1024). If omitted, BUFSIZE defaults to 1024.

IOBS

Specifies the number of input/output blocks to be created at initialization time for concurrent queue access. Value may be changed but it is recommended that the value supplied with the distributed version of the product (IOBS=5) be retained initially. As the number of active terminals increase, it may be necessary to increase the number of input/output blocks. If omitted, IOBS=3 is assumed.

RECORDS

Specifies the number of records per track for the Queue data sets. If omitted, RECORDS is automatically determined from the DASD device type. Value, if coded, must be one of the following:

- 11**
for 3330-(I or II)
- 15**
for 3350-type devices
- 25**
for 3375-type devices
- 31**
for 3380-type devices
- 33**
for 3390-type devices
- 28**
for 9345-type devices

SWFTRK | SCRTRK

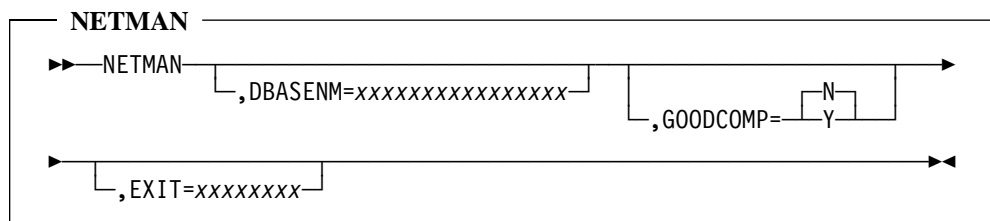
Specifies the maximum number of tracks to be assigned each scratch work file. Single tracks are allocated dynamically as needed, up to this maximum, from the scratch queue for scratch work files which are used for text editing and functions requiring internal sorts. Value if used, must be a number from 1 to 255. SWFTRK is optional. If omitted, the default is 10 tracks. The keyword SCRTRK can be used interchangeably with the keyword SWFTRK.

If this value is ever increased, be sure to verify and/or increase the allocation for the scratch queue. See the discussion of the SCRQ allocation under 3.2.4.4, "Space Requirements" on page 3-12.

5.2.28 NETMAN Statement

The NETMAN statement is required to use the realtime interface with CA-Netman.

5.2.28.1 Syntax



Where:

[illegible]

Specifies the 16-character name of the CA-Netman database associated with the copy of CA-Netman that is to receive the API transactions generated by CA-7 job completions. Embedded blanks are not allowed. This name is shown on the CA-Netman System Information (SYST) screen. See CA-Netman documentation for appropriate value.

GOODCOMP=N|Y

The value of this keyword is used to filter job completion data that is fed as input to the NETMAN interface.

N

Completion data is passed to the NETMAN interface for those jobs that abnormally terminate and for normally terminated jobs that have been restarted. This is the default.

Y

All data on all job completions (including jobs that complete normally) are passed to the NETMAN interface.

EXIT=xxxxxxxx

If CA-Netman is not used for problem tracking, this keyword may be used to identify a client-supplied interface module that will be called by the CA-7 problem management subtask to inspect job completion data.

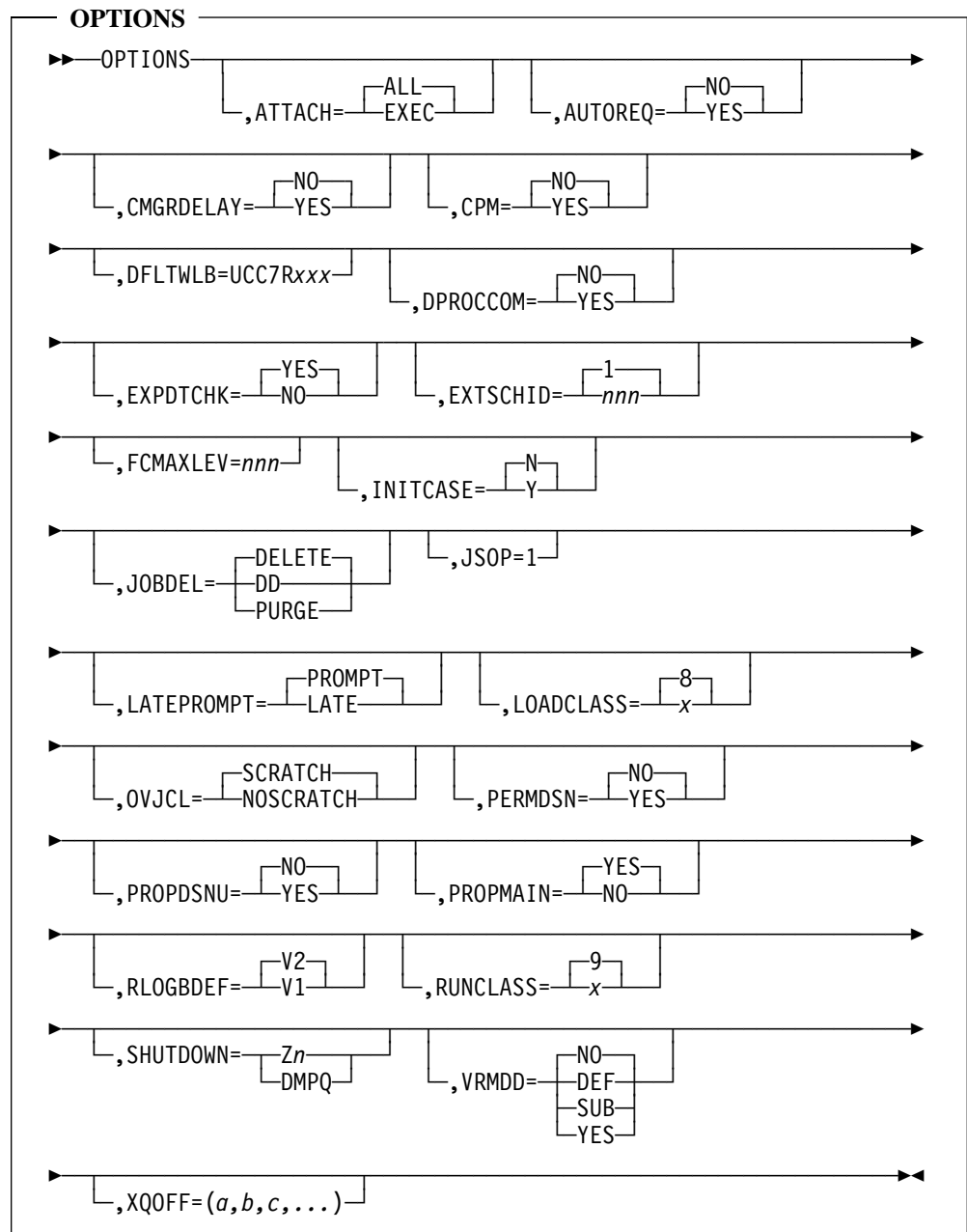
If CA-Netman is used for problem tracking, this keyword should be omitted.

For additional information on this exit, see 9.4.3, “Problem Management Interface” on page 9-37.

5.2.29 OPTIONS Statement

The OPTIONS statement can define a variety of processing options for the CA-7 system. The statement is optional. However, if the OPTIONS statement is coded, at least one keyword is required.

5.2.29.1 Syntax



Where:

AUTOREQ

Specifies whether CA-7 is to automatically REQUEUE jobs found to be active on a specific CPU when an IPL of that CPU occurs.

NO

CA-7 should not automatically requeue jobs. This is the default.

YES

CA-7 should automatically requeue jobs which are in the active queue, and which show they are running on the CPU which has been IPLed.

ATTACH

Specifies whether JCL attach processing should be attempted for nonexecutable jobs.

ALL

Indicates that all jobs, including nonexecutable jobs, should have JCL attach processing performed. This is the default.

EXEC

Indicates that only executable jobs should have JCL attach processing performed.

CMGRDELAY

Prompts to allow modify of the interval between CA-7 reads of the communications data set. If this keyword is not specified, the interval is 10 seconds.

Caution

This keyword should only be used in consultation with Computer Associates Technical Support. Inappropriate use of this option could have an adverse impact on CA-7 performance.

NO

CA-7 will wake up every 10 seconds to read the communications data set. This is the default.

YES

CA-7 will issue a message requesting confirmation of the interval between reads of the communications data set:

CA-7.IOPT - SHOULD CA-7 READ COMMDS EVERY FIVE SECONDS? REPLY TO CONFIRM (YES/NO)

Reply YES to confirm the interval. A reply of NO indicates that the duration of the interval will default to 10 seconds.

CPM

Used to activate the Critical Path Management interface. See CA-OPS/MVS II documentation for additional information on this feature.

NO

CA-7 will not activate the CPM interface. This is the default.

YES

CA-7 will activate the CPM interface. If this option is selected, CA-7 generates CPM tracking data for every job that is part of a FLOW. See the *CA-7 Interfaces Guide* for information on this interface.

When CPM=YES is specified, CA-7 ensures that RSRC=YES is coded on the DBASE statement. CA-7 also checks for the existence of a terminal that is defined as DEVICE=CCI. Both conditions must be satisfied to use CPM.

DFLTWLB

Used to specify a default workload balancing module to be loaded during CA-7 initialization. The WLB module must be located in a load library accessible to CA-7 during startup. The module name must meet the workload balancing module naming requirements of "UCC7Rxxx" where xxx is any valid user specified characters. If the DFLTWLB keyword is not specified, the default WLB module UCC7RDFL is used. This keyword is ignored for WARM and ERST starts of CA-7.

DPROCCOM

Indicates whether comments are to be ignored inside CA-DRIVER procedures.

NO

CA-DRIVER should perform variable substitution on procedure statements that begin with '/*'. Such statements will be included in CA-DRIVER expansions. This is the default.

YES

CA-DRIVER should not perform variable substitution on procedure statements that begin with '/*'. These statements will be ignored by CA-DRIVER and will not be included in any CA-DRIVER expansions.

nnnnnnnn

CA-DRIVER should not perform variable substitution on procedure statements that begin with the specified character string. These statements will be ignored by CA-DRIVER and will not appear in any CA-DRIVER expansions. The length of the character string must not exceed eight bytes and should not include blanks or commas. The value also cannot be one of the following: Y, YES, N, or NO.

EXPDTCHK

Used to specify whether CA-7 should check for an expiration date before updating a data set.

YES

CA-7 does not attempt to update the data set if the expiration date has not yet been reached. The default is YES.

NO

CA-7 does not check the expiration date before attempting the update.

Note: You should check with your systems programmer before overriding the default for this option. If you specify EXPDTCHK=NO, an operator request (WTOR) may be issued asking if CA-7 should be allowed to update the data set when a save or replace is performed on it (see IBM message IEC507D). The entire CA-7 address space may stop, waiting for the operator to answer the query.

EXTSCHID

Used to specify a global default schedule ID for externally tracked tasks. If an externally tracked task is not explicitly assigned a nonzero schedule ID, it is assigned this value. See 3.8, "Tracking External Tasks" on page 3-70 and 3.8.4, "\$L2EXTT Macro" on page 3-73 for more information.

The specified value can range from 1 to 255. The default value is 1.

FCMAXLEV

Allows the specification of the maximum number of levels that a forecast command (including FSTRUC) goes through before issuing an error message and terminating the forecast. This can help prevent loops in triggering structures.

The specified value can range from 1 to 999.

If FCMAXLEV is not specified, then the default maximum level is 99. The maximum level on individual commands can be overridden by using the LVL= keyword on the command itself.

INITCASE

Determines character translation mode in the CA-7 Editor. If Full Edit Mode is used, then:

N

Forces the translation of all data to uppercase. This is the default.

Y

Translation is controlled by case setting commands in the CA-7 Editor.

See the discussion of the FEM subcommands MIXED and UPPER in the *CA-7 Database Maintenance Guide* for further information.

JOBDEL

Allows the specification of how the DELETE function on the CPU Job Definition screen (DB.1) should be interpreted. The default is DELETE which means it will be interpreted as a normal delete. If this option is set to DD or PURGE, then when a user enters 'DELETE' as a function on the DB.1 screen it will be interpreted as if a DD or PURGE had been entered.

JSOP=

Specifies job submission options.

1 - If 1 is specified then CA-7 will not move a job from the request queue to the ready queue if there is another job with the same name in the request queue in restart status.

This option ensures that a job be complete (successfully completed, canceled or completion forced, and so on) before another job of the same name can be moved to the ready queue.

This may prove useful in certain situations involving the use of VRM resources.

For example, suppose JOBA is defined to use resource RSC1. Further assume that the RM.1 definition specifies TYPE=EXC and FREE=Y. If JOBA (say, job number 1) abends and is waiting in the request queue for restart, another instance of JOBA (job number 2) may move to the ready queue for submission. However, JOBA (1) still has exclusive control of the RSC1 resource that was acquired when it was initially submitted. Hence, JOBA (2) will not submit. Instead it will wait in the ready queue for the RSC1 resource to become available. An LQ display will show that the status of JOBA (2) is W-RSRC.

To allow JOBA (2) to run, one must take manual corrective action such as canceling or forcing completion of JOBA (1). JOBA (1) cannot be submitted while JOBA (2) is in the ready queue.

Another alternative is to use JSOP=1. If this option is used, then JOBA (2) will not be moved to the ready queue until JOBA (1) is no longer in the request queue.

LATEPROMPT

Specifies how the PROMPTS field on the DB.1 (JOB) panel is to be interpreted.

PROMPT

Indicates the traditional meaning for the field. This is the default.

LATE

Indicates that jobs which are defined with PROMPT=NO will never be marked as LATE on an LQ or LRLOG display regardless of how long they stay in the queues.

Jobs defined with PROMPT=YES are processed the same regardless of the setting for LATEPROMPT.

LOADCLASS

Specifies the default Workload Balancing (WLB) class for jobs brought into the system by the LOAD/H command. The value must be a single alphanumeric character. The default is 8.

OVJCL

Specifies whether the JCL members for jobs taken from the JCL override library should be deleted when the job completes successfully. If used the value must be one of the following:

SCRATCH

Indicates the override JCL members should be deleted upon successful completion of the job. This is the default.

NOSCRATCH

Indicates the override JCL members should not be deleted upon successful completion of the job.

PERMDSN

Used to specify if all data sets added to the CA-7 database should default to permanent. The default is NO. If you specify PERMDSN=YES, then ALL jobs added through LOAD processing and through the DB.6 panel default to PERM. The DB.3.1 screen shows a PERM value of N. However, the job does not consider the data set as a requirement since the DB.6 screen shows the PERM status.

PROPSNU

Used to specify if the user field of JQREC (JQUSER) should be propagated across data set triggers; that is, to copy the JQUSER field of the job creating/updating a data set to job(s) which are triggered by that data set. The default is NO.

PROPMAIN

Used to specify if job mainid propagation is to take place for triggered jobs. If you do not want a job's mainid propagated to any subsequently triggered jobs specify NO. The default is YES.

RLOGBDEF

Used to specify the format of LRLOG output when the command is entered through a batch terminal (BTI), and a specific format has not been specified on the command. If you have existing batch processes which are dependent on the pre-Version 3.2 format of the LRLOG display and you cannot change the command input, use this option to set up a default for ALL batch LRLOG commands. See LRLOG in the *CA-7 Commands Guide* for more information on the LRLOG command.

V2

Indicates that batch LRLOG commands should default to the current display format (Version 3.2+). This is the default.

V1

Indicates that batch LRLOG commands should default to the pre-Version 3.2 display format.

RUNCLASS

Specifies the default Workload Balancing (WLB) class for jobs brought into the system by the RUN/H command. The value must be a single alphanumeric character. The default is 9.

SHUTDOWN

Indicates the default CA-7 shutdown type that will be performed when the /SHUTDOWN command is issued. The default shutdown type is only activated if the /SHUTDOWN command is issued without parameters.

Zn

Specifies the method of shutdown as follows:

Z1

Specifies a "fast" shutdown of CA-7. Messages are not sent to the individual terminals and CA-7 does not wait for them to log off, but waits for batch terminals to complete.

Z2

Specifies that shutdown is to occur even if batch terminals are active but waits for online terminals to log off.

Z3

Specifies that shutdown occurs even if online and/or batch terminals are active.

Z4

A combination of Z1 and DMPQ.

Z5

A combination of Z3 and DMPQ.

Note: If a form of Zn is not specified, CA-7 does not terminate until all terminals have logged off. No batch terminals can be active.

DMPQ

Specifies that the CA-7 queues, except the SCRQ and DQTQ are to be unloaded during the shutdown process. These queues are unloaded to the CA-7 queue dump file. If only DMPQ is specified, CA-7 does not terminate until all batch terminal processing is completed and all online terminals are logged off.

Note: The VRM database component contains "active" resource records which relate to jobs currently in the CA-7 queues. These active records are dumped to the VRM dump queue file ddname CA7VDMP and are date/time stamped to match the queue record(s) to ensure synchronization during MOVQ processing.

VRMDD

Specifies whether to activate VRM device definition. For more information on the VRM device facility, see VRM Device Control in Chapter 5 of the *CA-7 Database Maintenance Guide*.

NO

This is the default. VRM device definition is not activated. No VRM device definitions are built. If any are detected on the VRM database, they are ignored when jobs are submitted.

DEF

Indicates that VRM device definitions are to be built when jobs are LOAded. However, any definitions are ignored at submission time. This option may prove useful while testing so that device definitions can be examined prior to being used in a production environment.

This option also requires that a SASSDTAB table be in a load library that can be accessed by CA-7.

SUB

Indicates that VRM Device Definitions will be used at job submission, however no definitions will be dynamically built when jobs are LOAded.

This option requires that RSRC=YES be coded on the DBASE statement. This option also requires that a SASSDTAB table be in a load library CA-7 can access.

YES

Combines the DEF and SUB options. If VRMDD=YES is coded, VRM Device Definitions are built dynamically when a job is LOAded, and those definitions are honored when jobs are submitted.

This option requires that RSRC=YES be coded on the DBASE statement. This option also requires that a SASSDTAB table be in a load library that CA-7 can access.

XQOFF

Specifies line function characters which should be disabled on the QM.1-X and QM.1-M screens (XQ and XQM). The characters that can be specified in the XQOFF list must correspond to valid XQ line functions. Valid line functions are: C, F, H, J, P, Q, R, S, U, V, X, and E. See the *CA-7 Commands Guide* for information on the XQ and XQM screens and their associated line functions.

If more than one function character is specified the list must be enclosed in parentheses with each character separated by commas. For example:

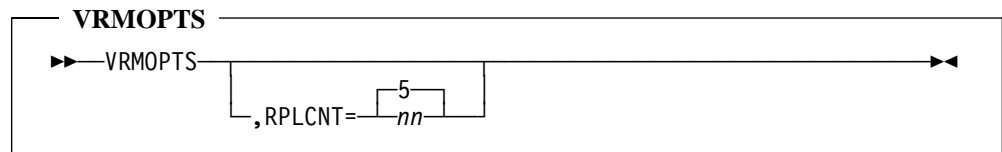
XQOFF=C	Disables the Cancel command on the XQ and XQM screens
XQOFF=(C,H)	Disables the Cancel and Hold commands on XQ and XQM

Note: Line functions which are disabled by the XQOFF parameter are not available to ANY CA-7 user regardless of security definitions. If users attempt to use one of these commands, they receive a message indicating the command has been disabled by site option. The XQOFF option does NOT affect any equivalent top line commands.

5.2.30 VRMOPTS Statement

The VRMOPTS statement can define processing options for the CA-7 Virtual Resource Management facility (VRM). The VRMOPTS statement is optional. However, if VRMOPTS is used, the RPLCNT keyword is required.

5.2.30.1 Syntax



Where:

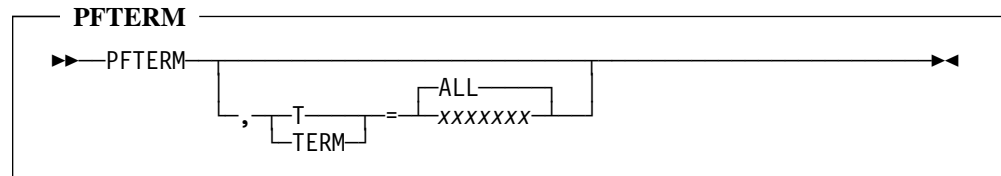
RPLCNT

Specifies the number of RPLs to be allocated for access to the CA-7 VRM data set. This establishes the number of threads that can access the data set concurrently. The specified value can range from 1 to 64, however, it is recommended that you specify at least 3. The default is 5.

5.2.31 PFTerm Statement

The PFTERM statements identify terminals for assigning PF/PA key assignments with PFnn and PAnn statements. These statements allow assignment by individual terminals at initialization. A PFTERM statement for a specific terminal must immediately precede the PF/PA statements for that terminal in the initialization file.

5.2.31.1 Syntax



Where:

TERM|T

Specifies the name of a terminal for which PF/PA key assignments are to apply. Value must be a 1- to 7-character terminal name as defined by NAME on the TERM statement. TERM is optional with a default of ALL. T may be used instead of TERM. The ALL option causes the PF/PA assignments which follow this statement to apply to all terminals not otherwise defined by a PFTERM statement.

Individual terminal assignments using the combination of PFTERM, PFnn, and PAnn statements must precede the PFTERM for T=ALL.

5.2.31.2 Examples

```

PFTERM,T=term1
PFnn statements for terminal 1
PAnn statements for terminal 1
PFTERM,T=term2
PFnn statements for terminal 2
PAnn statements for terminal 2
.
.
.
PFTERM,T=ALL
PFnn statements for all other terminals
PAnn statements for all other terminals

```

Note: See the /PROF command in the *CA-7 Commands Guide* for information on saving PA/PF key definitions for individual users.

5.2.32 PFnn and PAnn Statements

The PFnn statements define the program function (PF) key assignments for the terminals. The PAnn statements define the program access (PA) key assignments. One statement is required for each key to be assigned. These statements must immediately follow the PFTERM statement which defines the terminal(s) to which these assignments apply.

5.2.32.1 Syntax



Where:

nn

Specifies the key number in two digits 01 to 24 for PFnn; 01 to 03 for PAnn.

X...X

Represents a character string assigned to the key. This character string may be any commonly used top line command including the /ECHO command.

5.2.32.2 Examples

```
PF07,LQ
PF08,LPRE
PF09,/ECHO,M=(LQ,JOB=#)
.
.
.
PA01,GRAPHJ,ID=0370
.
.
.
```

If a key is defined with the `/ECHO` command, the command is displayed back to the terminal and the cursor is positioned at the `#` when the key is pressed.

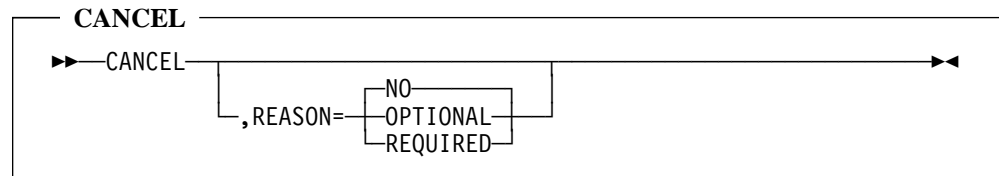
Note: The # must be the last character of the command for the cursor to position itself there.

The /PA and /PF commands may be entered at a terminal to override any values assigned here. Online menus and their related formatted and display screens temporarily use **PF3**, **PF7**, and **PF8** for specific purposes no matter what values are assigned here. Enter a /DISPLAY,ST=KEY command to view assigned values.

5.2.33 CANCEL Statement

This statement indicates if some type of explanation is to be allowed or required when a CPU job is manually canceled.

5.2.33.1 Syntax



Where:

Reason

Indicates whether a reason is required when manually canceling a job.

NO

Indicates that a reason may not be provided. This is the default. With this option, there is no opportunity to change your mind after you have requested a C function on the QM.1 screen.

OPTIONAL

Indicates that a reason may be provided but is not required. When specified:

- for each C function on a QM.1 screen, transfer is made to the REASON FOR CANCEL screen. That screen allows the user to provide a brief explanation of why the job is to be canceled. See the QM.1 screen for further discussion. Just displaying the REASON FOR CANCEL screen also provides an opportunity for the user to reconsider whether the job should be canceled before doing so.
- for each top line CANCEL command, the REASON keyword can be used to provide a brief explanation of why the job is to be canceled. See that command for further discussion of that keyword and its use.

REQUIRED

Indicates that a reason must be provided before the cancellation can occur.

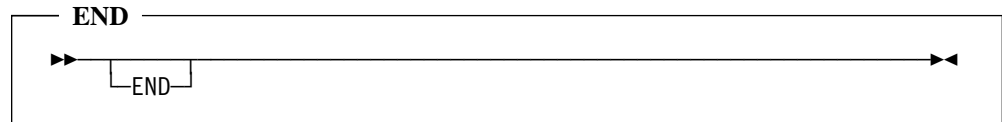
When specified:

- for each C function on a QM.1 screen, transfer is made to the REASON FOR CANCEL screen. That screen requires the user to provide a brief explanation of why the job is to be canceled. See the QM.1 screen for further discussion. Just displaying the REASON FOR CANCEL screen also provides an opportunity for the user to reconsider whether the job should be canceled before doing so.
- for each top line CANCEL command, the REASON keyword must be used to provide a brief explanation of why the job is to be canceled. See that command for further discussion of that keyword and its use.

5.2.34 END Statement

The END statement terminates the initialization file. It is optional but when used must be the last statement.

5.2.34.1 Syntax



There are no keywords. END must begin in column 1.

5.3 Offline Verification of Initialization File

The initialization file may require many statements depending on the user site and how CA-7 is being used there. The opportunity for an error to occur somewhere in a complete file increases with the number of statements.

Accuracy of the initialization file is not only a concern when building the file for the very first time. It is also a concern in an ongoing maintenance mode whenever changes become necessary.

To assist the user in verifying the accuracy of the initialization file, a batch edit function is provided. This job enables the user to verify the contents of the file without having to start up CA-7 for any other functions. It should be run anytime initialization file changes are made to verify accuracy before implementing the changes in a production mode. It is especially helpful when terminals are added to or deleted from the initialization file. This batch job can be run whether or not CA-7 is already active. The procedure discussed under 4.2, “Test Copy of CA-7” on page 4-8 is not necessary.

5.3.1 Batch Edit Execution

No changes are required within the initialization file to indicate that this is an initialization file edit. Once a successful edit has been done, the file can then be used "as is" to start up CA-7 in the desired way.

```
//CA7EDIT JOB.....,REGION=3072K
//*****
//*          EXECUTION OF CA-7 INITIALIZATION FILE EDIT      *
//*****
//EDIT      EXEC PGM=UCC7,PARM='TYPE=EDIT'
//STEPLIB   DD DISP=SHR,DSN=ca7.loadlib
//UCC7OUT    DD SYSOUT=*
//SYSUDUMP  DD SYSOUT=*
//UCC7IN     DD *
           *** CA-7 Initialization File goes here ***
/*
```

Figure 5-1. Sample Initialization File Edit JCL

This JCL executes PGM=UCC7, not a cataloged procedure, with a TYPE value of EDIT through the PARM value on the EXEC statement. This causes CA-7 to terminate after editing the initialization file, without attempting any of its other functions, no matter what values are coded within the file itself. The TYPE value in the INIT statement in the file should have whatever value is needed whenever the file is used in production. EDIT is not valid in the INIT statement. It is only valid as a PARM value. See 6.1, “CA-7 Execution” on page 6-2 for a discussion of acceptable values for TYPE in the INIT statement.

Any errors that are encountered within the file are handled as they are any other time that CA-7 is executed. That is, the messages and abend codes are the same here for any errors that are encountered.

Successful completion of the edit is indicated by the message:

CA-7.998 - INITIALIZATION FILE EDIT COMPLETE.

Since this edit is performed by CA-7 itself, the REGION size must be large enough to handle the environment defined by the initialization file. The REGION size needed for a production run of CA-7 is much larger than that needed for this run. This process only requires DD statements for the file input data set UCC7IN and the print data set UCC7OUT. Other DDs should be omitted from the JCL to avoid allocation conflicts with any production copy of CA-7. The regular execution JCL for CA-7 can be used only if CA-7 is not currently executing.

Note: If using JCLCHECK, then the SYSPROC DD statement is required (if used).

Not all initialization file problems can be detected by the TYPE=EDIT checking. For example, correspondences between TERM statements and STATIONS statements cannot be tested. Even so, TYPE=EDIT is effective for other types of conditions and is recommended for use when initialization file changes are made.

Chapter 6. Execution

This chapter discusses considerations for execution of CA-7 and ICOM.

6.1 CA-7 Execution

Module UCC7 is the central control entity of CA-7 and must be active whenever CA-7 functions are to be performed. Whenever CA-7 is to provide automatic scheduling, monitoring and control of work, the Independent Communications Manager (ICOM) must also be active. Execution of ICOM is explained in 6.6, “ICOM Execution” on page 6-26.

6.1.1 REGION Size

In a typical data center, CA-7 should be given a REGION size of at least 4M of virtual memory. When a larger number of terminals, user exits, and so forth require a larger region, the size is normally increased in 1/2 megabyte (512K) increments. CA-7 also uses region above 16M for database access as well as queue access. The amount of region above 16M varies based on initialization file options as well as usage of the trailer queue. Usually, a minimum of 4M should be available above 16M.

Additional region size requirements (over 3M below the 16M line) can be calculated using the following guidelines. Also see the CWORK PARM value discussed later.

1. If user exits are being used which use OS GETMAIN macros to acquire sizable amounts of memory, increase the REGION value by the appropriate amount.
2. If the CA-7 CA-JCLCheck interface is used, the REGION value should be increased by at least 512K.
3. If RESTART,ARF=YES or MAINT, increase the region size by at least 1024K.

For region requirements for memory-resident queues and the trailer queue, see the discussion of the INDEX parameter of the 5.2.26, “FORMAT Statement” on page 5-63.

6.1.2 PARM Values

The PARM data on the EXEC statement defines the amount of memory, from the CA-7 region, to be reserved for interface work areas within CA-7. Optionally, the user may also specify the startup type, with the TYPE keyword, as a PARM value.

The available PARM keyword parameters are:

TYPE=type

Specifies type of startup: COLD, WARM, EDIT, ERST, FORM, MOVQ, or DORM. See 5.3, “Offline Verification of Initialization File” on page 5-86 for EDIT, and 6.3.1, “Startup Options” on page 6-18 for the other options. The TYPE value specified here overrides the value specified in the initialization file INIT statement. However, a valid TYPE value must always be coded in the initialization file.

CABND=nnn

Specifies the number of 2K blocks of memory to be reserved for abend processing during initialization. The default value is 2 (4K of memory).

CWORK=nnn

Specifies the number of 2K blocks of memory to be reserved for initialization and interface work areas. The default is 110. Requirements for CWORK vary depending on the host operating system and REGION size for CA-7. The following discussion outlines those considerations.

Start with a value of 10 (20K of memory).

For each concurrent use of the CA-1 interface command TIQ, add 10 (20K of memory).

For each concurrent use of the CA-11 interface command ARTS, add 20 (40K of memory).

If Version 11 or later of CA-Panvalet is being used, add 150 (300K of memory).

If online user exits use OS GETMAIN macros to acquire storage, CWORK must be increased by the appropriate amount to reserve this memory.

If memory-resident indexes are being used for any queue(s) or if the entire queue(s) are to be memory-resident, as specified in the FORMAT statement(s), add an appropriate amount. See the discussion of the INDEX option of the FORMAT statement for the formula to determine the size of these area(s), divided by 2K and rounded to the next highest whole number. (See 5.2.26, “FORMAT Statement” on page 5-63.) For example, if the area size is 5804 bytes, additional CWORK needed would be $5804/2048=2.83$ or an additional 3 for CWORK.

If the CA-7 CA-JCLCheck interface is used, increase CWORK by 250.

6.1.3 Online Execution

To perform automatic scheduling and job submission functions, you must initialize CA-7 for online operation.

The CA-7 SYSGEN macros generate JCL necessary for executing CA-7 online. Sample JCL is given in CA-7 Online Execution JCL on page 6-5. For a description of the DD statements, see Table 6-1 on page 6-6. The SYSGEN macros also generate a sample initialization file for executing CA-7 online. You may need to modify this file to reflect your own environment. See Sample Online Initialization File on page 6-10 for a listing of a sample online initialization file.

The first online execution of CA-7 must be done with a TYPE=FORM start. This can be specified with the PARM information on the EXEC statement. After the first time, a TYPE=WARM or TYPE=ERST start should be done. For a more detailed description of the initialization file parameters, see 5.2, “Initialization File Statements and Keywords” on page 5-4 and following pages.

```

/*-----*
/*
/*          'SAMPLE EXECUTION OF CA-7 ONLINE'
/*
/*-----*
//jobname JOB.....,.....,REGION=4096K
//CA7ONL  PROC  CABND=4,CWORK=300,TYPE=WARM
//CA7     EXEC  PGM=UCC7,TIME=1440,
//          PARM='CABND=&CABND,CWORK=&CWORK,TYPE=&TYPE'
//STEPLIB DD    DISP=SHR,DSN=ca7.cailib
//SYSUDUMP DD   SYSOUT=A
//ABNLIGNR DD   DUMMY
//NOMDUMP DD   DUMMY
//UCC7DUMP DD   SYSOUT=*
//UCC7OUT DD    SYSOUT=A
//DBPARMS DD   DISP=SHR,DSN=allvsam.database.parms
/* remove these three DD statements if ALLOCDYN in DBPARMS
//UCC7JLIB DD   DISP=SHR,DSN=user.defined.job.dataset
//UCC7DLIB DD   DISP=SHR,DSN=user.defined.dataset.dataset
//UCC7IDS DD    DISP=SHR,DSN=user.defined.index.dataset
//CA7RSRC DD    DISP=SHR,DSN=user.defined.VRM.dataset
//CA7ARFDB DD   DISP=SHR,DSN=user.defined.ARF.dataset
//CARPROC DD    DISP=SHR,DSN=cadriver.proclib
//UCC7LOG DD     DISP=SHR,DSN=ca7.logp
//UCC7CMDS DD    DISP=SHR,DSN=ca7.commds
//UCC7SUB1 DD    DISP=SHR,DSN=ca7.submtds1
//UCC7IRD1 DD    SYSOUT=(A,INTRDR),DCB=BLKSIZE=80
//UCC7IRDR DD    SYSOUT=(A,INTRDR),DCB=BLKSIZE=80
//UCC7CKPT DD    DISP=OLD,DSN=ca7.ckpds
//UCC7SCRQ DD    DISP=OLD,DSN=ca7.scrque
//UCC7PREQ DD    DISP=OLD,DSN=ca7.preque
//UCC7REQQ DD    DISP=OLD,DSN=ca7.reqque
//UCC7RDYQ DD    DISP=OLD,DSN=ca7.rdyque
//UCC7ACTQ DD    DISP=OLD,DSN=ca7.actque
//UCC7PSTQ DD    DISP=OLD,DSN=ca7.pstque
//UCC7PRNQ DD    DISP=OLD,DSN=ca7.prnque
//UCC7TRLQ DD    DISP=OLD,DSN=ca7.trlque
//UCC7DQTQ DD    DISP=OLD,DSN=ca7.dqtque
//UCC7QDMP DD    DISP=OLD,DSN=ca7.dmpque
//CA7VDMP DD     DISP=SHR,DSN=ca7.ca7vdmp
//U7PACK50 DD    DISP=SHR,UNIT=3350,VOL=SER=PACK50 (site dependent)
//U7PACK75 DD    DISP=SHR,UNIT=3375,VOL=SER=PACK75 (site dependent)
//U7PACK80 DD    DISP=SHR,UNIT=3380,VOL=SER=PACK80 (site dependent)
//BTERM01I DD    DISP=SHR,DSN=ca7.batchi#1
//BTERM01O DD    DISP=SHR,DSN=ca7.batcho#1
//UCC7STAT DD    DISP=SHR,DSN=ca7.statfile
//BROWSE DD      DISP=SHR,DSN=ca7.browse
//UCC7WLP DD     DISP=SHR,DSN=user.defined.fwlp.dataset
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=80 * nnn)
//FREE       EXEC PGM=SASSICLR,COND=EVEN
//STEPLIB DD    DISP=SHR,DSN=ca7.loadlib
//UCC7CMDS DD    DISP=SHR,DSN=ca7.commds
//          PEND
//STEP1      EXEC CA7ONL
//CA7.UCC7IN DD *
/*          *** CA-7 online init file goes here ***
/*

```

Figure 6-1. CA-7 Online Execution JCL

6.1.3.1 CA-7 DD Statements

Table 6-1 (Page 1 of 4). CA-7 DD Statements		
DDNAME	Explanation	Status
STEPLIB	CA-7 load library. If the base calendars are stored in a separate data set, another DD defining the calendar library data set must be concatenated with CA-7 LOADLIB. The CA-1 LOADLIB must be concatenated if using TIQ. The CA-11 LOADLIB must also be concatenated if using ARTS.	Required unless link listed.
SYSUDUMP	Standard SYSUDUMP data set (SYSOUT=A).	Required for debugging purposes.
UCC7OUT	Output data set for listing the CA-7 initialization file.	Required.
DBPARMS	Parameters to describe a CA-7 database.	Required.
UCC7JLIB	Job data set of the CA-7 database.	Required unless you specify ALLOCDYN in DBPARMS.
UCC7DLIB	Dataset data set of the CA-7 database.	Required unless you specify ALLOCDYN in DBPARMS.
UCC7IDS	Index data set of the CA-7 database.	Required unless you specify ALLOCDYN in DBPARMS.
CA7RSRC	VRM resource database.	Required.
CA7ARFDB	Automated Recovery Facility Database.	Required if RESTART, ARF=YES or ARF=MAINT.
UCC7LOG	Required CA-7 log data set. Specify primary log data set name.	Required.
UCC7CMDS	Communications data set. In multi-CPU environments, must reside on a shared DASD device available to all CPUs which process CA-7 controlled jobs.	Required for online. Optional for batch.

Table 6-1 (Page 2 of 4). CA-7 DD Statements

DDNAME	Explanation	Status
UCC7SUBn or UCC7IRDn.	Submit data set used for submitting JCL to the host system. In a multi-CPU environment each host system requires its own submit data set. Shared spooled systems require only the UCC7IRDn DD. The n is used to provide ddname uniqueness.	Required for online. Not used by batch. UCC7IRDR is required if RUN command on DB.7 screen is to be used.
UCC7CKPT	Checkpoint data set.	Required.
UCC7SCRQ	Scratch queue.	Required.
UCC7PREQ	Preprocess queue.	Required for online. Optional for batch.
UCC7REQQ	Request queue.	Required for online. Optional for batch.
UCC7RDYQ	Ready queue.	Required for online. Optional for batch.
UCC7ACTQ	Active queue.	Required for online. Optional for batch.
UCC7PSTQ	Postprocess queue.	Required for online. Optional for batch.
UCC7PRNQ	Prior-run queue.	Required. See Note 1.
UCC7TRLQ	Trailer queue.	Required. See Note 1.
UCC7DQTQ	Disk queue table queue.	Required. See Note 1.
UCC7QDMP	Queue dump data set.	Optional. Required only for DMPQ, Z4, and Z5 shutdown options.

Table 6-1 (Page 3 of 4). CA-7 DD Statements		
DDNAME	Explanation	Status
CA7VDMP	VRM dump data set.	Optional. Required only for DMPQ, Z4, and Z5 shutdown options.
JCLnnn	CA-Librarian or CA-Panvalet JCL data set.	Optional. See CA-Librarian and CA-Panvalet in the <i>CA-7 Interfaces Guide</i> .
U7volser	Identifies the volume on which data sets reside that are to be accessible to CA-7. The volser in the ddname and the VOL=SER= parameter must match and must specify the actual volume serial number.	Required for both online and batch for all volumes containing data sets to be accessed by CA-7, unless dynamic allocation is used.
BTERMnnI	Input data set for batch terminal nn.	Optional for online. Required for batch. See Note 2.
BTERMnnO	Output data set for batch terminal nn.	Optional for online. Required for batch. See Note 2.
UCC7STAT	Statistics data set	Required.
BROWSE	A sequential DASD browse data set which contains records of print image messages produced for station name MASTER.	Optional. ddname must match GROUP statement NAME parameter in the initialization file.
UCC7WLP	Workload planning data set	Required for FWLP.
UCC7IN	Input data set for the initialization file	Required.

Table 6-1 (Page 4 of 4). CA-7 DD Statements		
DDNAME	Explanation	Status
CARPROC	CA-Driver procedure library.	Optional for online only. Used by the CA-Driver component.
<p>Note 1: For batch execution of CA-7, these DD statements may be omitted if their FORMAT statement in the initialization file points to the UCC7SCRQ data set.</p> <p>Note 2: The first seven characters of these ddnames must match the initialization file GROUP statement NAME parameter. The first seven characters <i>do not</i> have to be BTERMnn; however, the last (8th) character must be either an I or O (alpha) to indicate either an <u>i</u>nput or <u>o</u>utput data set.</p>		

```

*-----*
*                                     *
*          SAMPLE INITIALIZATION FILE FOR CA-7 ONLINE          *
*                                     *
*-----*
RESIDENT,APGPL=204800,NETWORK=ALL
CUST,ID=***** YOUR COMPANY NAME GOES HERE *****
INIT,TYPE=WARM,CONFIG=(MVS,JES2),RUNOPT=(NSTA),PERFORM=(4,5,6,7)
SVCNO,SASSVC=YES
APPLCTN,NAME=SASSPROG,ATTR=LOAD
FMTBLK,NAME=SASSFMTH,ATTR=LOAD
FMTBLK,NAME=SASSFMTH,ATTR=LOAD
CALBLK,NAME=SCAL00PE
CALBLK,NAME=SCAL0003
UCC7VTAM,APPL=CA7,VTNAME=VTM
GROUP,NAME=VGROUP,DEVICE=3270V,LNAME=(HLINE,VLINE),
    OPEN=YES,VLOGOFF=PF12
LINE,NAME=HLINE,TNAME=HTERM,OPEN=YES
TERM,NAME=HTERM,DEVICE=3270V,CONS=MASTR,VTAMID=VTAMTERM
LINE,NAME=VLINE,TNAME=VTM#010,OPEN=YES
TERM,NAME=VTM#010,DEVICE=3270V,CONS=ALTRN,SIMLOG=NO
GROUP,NAME=CONSOLE,DEVICE=CONSL,LNAME=CONSOLE,OPEN=YES
LINE,NAME=CONSOLE,TNAME=CONSOLE,BUFSIZE=150,OPEN=YES
TERM,NAME=CONSOLE,DEVICE=CONSL,LINLEN=110,NLINE=10,CONS=ALTRN
GROUP,NAME=BROWSE,DEVICE=BSAM,LNAME=BROWSE
LINE,NAME=BROWSE,TNAME=BROWSE,BUFSIZE=800
TERM,NAME=BROWSE,DEVICE=BSAM,LINLEN=80,NLINE=24
GROUP,NAME=BTERM01,DEVICE=BATCH,LNAME=BLINE1
LINE,NAME=BLINE1,BUFSIZE=3120,TNAME=BTERM1
TERM,NAME=BTERM1,DEVICE=BATCH,NLINE=60,LINLEN=137
GROUP,NAME=TRAILER,DEVICE=TRLDV,LNAME=TLINE,OPEN=YES
LINE,NAME=TLINE,BUFSIZE=1024,TNAME=TTERM,OPEN=YES
TERM,NAME=TTERM,DEVICE=TRLDV,NLINE=1,LINLEN=80
INIT2
SECURITY,NAME=SASSSECI
STATIONS,TRMID=BROWSE,STANIDS=MASTER
STATIONS,TRMID=HTERM,STANIDS=(HTERM1,HTERM2)
STATIONS,TRMID=CONSOLE,STANIDS=(MVS,MCO)
STATIONS,TRMID=BTERM1,STANIDS=(BTI1)
STATIONS,TRMID=TTERM,STANIDS=(TRAILER)
DBASE,JOB=CA07LOGP,PROCLD=CA7LOAD,RSRC=YES
ALOG1,DSN=CAI.CA7.LOGP
ALOG2,DSN=CAI.CA7.LOGS
RESTART,REASON=YES
SCHEDULE,HIJBNUM=1000,SCNSPAN=4,SCNINCR=2,QDWELL=30,REPROMPT=10
CPU,DDNAME=UCC7IRDR,BARRIER=20,HOST=JES2,MAINIDS=(SY0,SY*)
JCL,DSN=CAI.CA7.HELP,INDEX=255,DYN=2
JCL,DSN=CAI.CA7.JCLLIB,INDEX=200,DYN=2
JCL,DSN=CA7.JCLDS.JCL0,INDEX=0,DYN=2
FORMAT,DDNAME=UCC7SCRQ,DISP=0
FORMAT,DDNAME=UCC7PREQ,DISP=0,FLNAME=UCC7PREQ
FORMAT,DDNAME=UCC7REQQ,DISP=0,FLNAME=UCC7REQQ
FORMAT,DDNAME=UCC7RDYQ,DISP=0,FLNAME=UCC7RDYQ
FORMAT,DDNAME=UCC7ACTQ,DISP=0,FLNAME=UCC7ACTQ
FORMAT,DDNAME=UCC7PSTQ,DISP=0,FLNAME=UCC7PSTQ
FORMAT,DDNAME=UCC7PRNQ,DISP=0,FLNAME=UCC7PRNQ
FORMAT,DDNAME=UCC7TRLQ,DISP=0,FLNAME=UCC7TRLQ
FORMAT,DDNAME=UCC7DQTQ,DISP=0,FLNAME=UCC7DQTQ
DAIO,IOPS=5,DASD=3380
END

```

Figure 6-2. Sample Online Initialization File

6.1.4 Batch Execution

CA-7 may be executed in batch-only mode for maintenance or reporting activities which do not require automatic scheduling and job submission facilities or online terminals. This type of execution should be run only when CA-7 is first installed using the installation job N220.

Note: CA-7 cannot be executed in batch and online mode at the same time. The batch terminal interface facility can be used for batch maintenance and reporting while CA-7 is executing online. See the batch terminal interface in the *CA-7 Commands Guide*.

JCL necessary for executing CA-7 in batch-only mode is generated by the CA-7 SYSGEN macros. For a description of the required DD statements see Table 6-1 on page 6-6. Sample JCL for batch execution is given on 6-13. A sample initialization file for in batch-only mode is also generated by the SYSGEN macros. See 6-14 for a listing of a sample batch-only initialization file. For batch-only execution of CA-7, ICOM is not required. Also, batch-only execution must never be attempted while CA-7 is active online.

The following are guidelines for initialization file changes for batch-only. No changes should be needed to the sample batch initialization file produced during SYSGEN unless the batch terminal ddname has changed.

6.1.4.1 RESIDENT Statement

Specify NETWORK=BAT

6.1.4.2 INIT Statement

RUNOPT=SCAN is not allowed.

Specify RUNOPT=REPT and TYPE=COLD to perform database maintenance and reporting. For this operation, the queues may be allocated to temporary data sets and the communications data set is not required.

6.1.4.3 GROUP Statement

Remove all GROUP statements which do not specify DEVICE=BATCH. Only one is needed.

6.1.4.4 LINE/TERM Statement(s)

Remove all LINE and TERM statements belonging to GROUP statements which have been removed. Only one LINE or TERM statement is needed.

Note: There must be GROUP/LINE/TERM statement sequences for the batch terminal.

6.1.4.5 SECURITY Statement

Ensure that the Security module includes SECURITY macro definitions for the specified batch terminal.

6.1.4.6 STATIONS Statement

Remove STATIONS statements for terminals no longer defined due to removal of GROUP/LINE/TERM statements. The batch terminal should specify STANIDS=MASTER.

6.1.4.7 ALOG1/ALOG2 Statements

DSN= may be indicated as temporary (for example, &&LOGP).

If the log data set is assigned to a tape for batch runs, the ALOG2 statement must be removed. The ALOG2 statement is never needed.

6.1.4.8 SCHEDULE Statement

Not used.

6.1.4.9 CPU Statement

Not used.

6.1.4.10 FORMAT Statement

For RUNOPT=REPT, remove FORMAT statements except for UCC7SCRQ, UCC7DQTQ, UCC7PRNQ, and UCC7TRLQ. DISP=F must be used if they are on temporary disk space. The ddname on these four format statements may specify UCC7SCRQ. FLNAME must be coded on the DQT, PRN, and TRL queue FORMAT statements. The DD in the batch execution JCL for the UCC7SCRQ may then specify a temporary file.

```

/*-----*
/*
/*          'SAMPLE EXECUTION OF CA-7 BATCH.'
/*
/*-----*
//jobname JOB .....REGION=3072K
//CA7BAT  PROC  U=SYSDA,CABND=1
//CA7     EXEC  PGM=UCC7,
//          PARM='CABND=&CABND'
//STEPLIB DD DISP=SHR,DSN=ca7.loadlib
//SYSUDUMP DD SYSOUT=A
//UCC7OUT DD SYSOUT=A
//DBPARMS DD DISP=SHR,DSN=ca7.database.parms
//UCC7JLIB DD DISP=OLD,DSN=user.defined.job.dataset
//UCC7DLIB DD DISP=OLD,DSN=user.defined.dataset.dataset
//UCC7IDS  DD DISP=OLD,DSN=user.defined.index.dataset
//CA7RSRC DD DISP=OLD,DSN=user.defined.VRM.dataset
//UCC7LOG DD DSN=&TMPLOG1,UNIT=&U,DISP=(,PASS),
//          DCB=(RECFM=VB,LRECL=1400,BLKSIZE=1404),
//          SPACE=(TRK,80,,CONTIG)
//UCC7CKPT DD UNIT=&U,SPACE=(CYL,1)
//UCC7SCRQ DD UNIT=&U,SPACE=(TRK,100,,CONTIG)
//U7PACK50 DD DISP=SHR,UNIT=3350,VOL=SER=PACK50      (site dependent)
//U7PACK75 DD DISP=SHR,UNIT=3375,VOL=SER=PACK75      (site dependent)
//U7PACK80 DD DISP=SHR,UNIT=3380,VOL=SER=PACK80      (site dependent)
//BTERM010 DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=552)
//          PEND
//STEP1 EXEC UCC7BAT
//CA7.UCC7IN DD *
***  batch init file  ***
//CA7.BTERM01I DD *
***  CA-7 commands go here  ***
/*

```

Figure 6-3. Batch-Only JCL

```

*-----*
*                                     *
*          'SAMPLE INITIALIZATION FILE OF CA-7 BATCH.'          *
*                                     *
*-----*
RESIDENT,APGPL=51200,NETWORK=BAT
CUST,ID=***** YOUR COMPANY NAME GOES HERE *****
INIT,TYPE=COLD,CONFIG=(MVS,JES2),RUNOPT=(NSTA,NCWT,REPT)
SVCNO,SASSVC=NO
APPLCTN,NAME=SASSPROG,ATTR=LOAD
FMTBLK,NAME=SASSFMTA,ATTR=LOAD
CALBLK,NAME=SCALxyy      (site dependent)
CALBLK,NAME=SCALxyy      (site dependent)
GROUP,NAME=BTERM01,DEVICE=BATCH,LNAME=HLINE1,OPEN=YES
LINE,NAME=HLINE,OPEN=YES,BUFSIZE=3120,TNAME=HTERM1
TERM,NAME=HTERM1,DEVICE=BATCH,NLINE=60,LINLEN=137
INIT2
SECURITY,NAME=SASSSECI
STATIONS,TRMID=HTERM1,STANIDS=(LTERM1,MASTER)
DBASE,DB=ALLVSAM
ALOG1,DSN=*&TMPLOG1
JCL,DSN=USER.JOBLIB,INDEX=0
JCL,DSN=SYS1.JOBLIB,INDEX=1,ALT=0
JCL,DSN=CA-7.HELP,INDEX=255
FORMAT,DDNAME=UCC7SCRQ,DISP=F
FORMAT,DDNAME=UCC7SCRQ,DISP=F,FLNAME=UCC7DQTQ
FORMAT,DDNAME=UCC7SCRQ,DISP=F,FLNAME=UCC7PRNQ
FORMAT,DDNAME=UCC7SCRQ,DISP=F,FLNAME=UCC7TRLQ
DAIO,BUFSIZE=1024,IOBS=5,RECORDS=15,DASD=3350
END

```

Figure 6-4. Sample Batch-Only Initialization File

6.2 Initialization Considerations

Initialization options are determined by the PARM values on the EXEC statement of CA-7 and the values coded in the initialization file. The initialization file (INIT file) is accessed by CA-7 through the UCC7IN DD statement in CA-7 JCL. Before these values are coded, the user should give some thought and analysis to the following items:

- terminals to be used by CA-7
- system startup options desired
- JCL data sets available to CA-7

CA-7 is usually initialized in batch mode just one time, during the initial installation. After installation, the online version is used.

Other initialization considerations are:

6.2.1 Date and Time Validation Option

The date and time validation feature provides the opportunity to verify and manually correct system clocks on all CPUs whenever CA-7 is initialized. This feature can be used to ensure that the date and time are correct within the system. Schedule scan produces improper results if those values are incorrect.

An initialization file parameter keyword, `VALIDATE`, is used to specify that validation is not to be performed or to designate a tolerance value for disabling the validation process. The time tolerance is based on the difference (in hours) between the current system date/time values and the most recent CA-7 checkpoint date/time values (if available).

The system initiates the verification process by displaying the current date and time values followed by a WTOR requesting action. A response to the WTOR indicates the desired action of CA-7. When the WTOR is issued, the following options are available:

- Manually correct the system clocks before proceeding. However, if the system clocks are acceptable, clock adjustment is not necessary.
- Proceed with schedule scan deactivated.

Whenever the desired option has been entered, the current date and time are displayed again, after the operator response, before proceeding. Any manual adjustments to the date or time while the WTOR was outstanding are obvious from the before and after displays.

6.2.2 Log Usage

CA-7 automatically logs:

- transactions from terminals
- internal events
- queue processing
- SMF feedback
- messages for the master station

Log data sets can be allocated on DASD or tape. If tape is selected, only one log data set is required, and a tape drive must be dedicated to CA-7.

The recommended configuration for the log file is to designate two disk data sets. This way, when one file is full, a switch is made to begin recording in the other log file. This is similar to how SMF handles logging. With CA-7, the job to dump the data is also automatically submitted whenever the switch occurs.

Only the UCC7LOG DD statement, pointing to the primary log file, is needed. The secondary log file must reside on the same pack as the primary file, with a different data set name but does not require a DD statement. The secondary log file data set name is used in combination with the ddname UCC7LOG to access the data set.

The initialization file must specify both the primary and secondary log data set names (ALOG1 and ALOG2 statements) plus the name of the job which dumps the primary log file (DBASE statement JOB parameter). That job name must be defined in the CA-7 database and the JCL for the job must reside in a JCL data set accessible to CA-7.

6.2.3 Batch Job vs. Started Task

CA-7 can run as a started task or a batch job. ICOM can also be a started task. Regardless of whether CA-7 is a batch job or a started task, it processes exactly the same way.

6.3 Startup Procedures

An initialization process must be performed to start CA-7 functions that schedule and control the production environment. This is controlled by the CA-7 initialization file. Statements within the file define the following:

- environment in which the Central Control System resides
- processing options to be exercised
- terminal and security definitions to be used
- type of initialization to be performed

An initialization process is necessary regardless of whether CA-7 is to be used in batch-only mode for maintenance type functions, or in online mode for automatic scheduling and production control.

Both batch and online activities use the same programs. User options in the initialization file define the operating environment and the functions over which it is to have control. The file is processed when CA-7 begins execution.

You must also activate ICOM, the Independent Communications Manager. ICOM, like CA-7, is activated through JCL. ICOM must be given a high priority. In a multiple CPU environment, one host CPU has both CA-7 and ICOM active. All other CPUs to which CA-7 submits jobs only have ICOM active.

You must make certain data sets associated with the CA-7 system available in addition to activating CA-7 and ICOM. The batch and online execution topics identify them.

The following considerations help to simplify the initialization process:

- Facilities such as the partitioned data sets, CA-7 editor/TSO/CA-Roscoe, CA-Panvalet/CA-Librarian, and so forth, can be used for storing and accessing CA-7 and ICOM JCL and the CA-7 initialization file.
- Separate JCL members should be established for batch and online JCL.
- Separate initialization files must be set up for batch-only and online execution.
- If the CA-7 JCL is stored in a location other than PROCLIB, the initialization file to be used can be in-stream with the JCL.
- Comment statements may be included in the initialization files to document usage of each file or portion of a file. Comments are indicated by an asterisk (*) in column 1.
- For nonshared spool environments, separate ICOM JCL members should be established for each CPU. These members should be stored in libraries accessible to the CPU on which each is to be run.

6.3.1 Startup Options

There are seven types of initialization or startup options available for restarting CA-7. These are:

1. WARM start
2. ERST start
3. COLD start
4. MOVQ start
5. FORM start
6. DORM start
7. EDIT start

The startup option is specified in the INIT statement of the initialization file or as a PARM in the EXEC statement; but if specified as a PARM in the EXEC statement, it overrides whatever is specified in the INIT statement. In any case, a TYPE parameter must be specified in the INIT statement even when a TYPE is included in the PARM data.

The option has no default value and must be coded in the following format:

TYPE=type

The acceptable values and their meanings are discussed in detail in the following materials.

The normal restart procedure is a WARM start. If CA-7 will not activate, try a specified ERST (emergency restart) start. If CA-7 will still not activate, then your CA-7 coordinator should be contacted. It may be necessary to do a COLD start.

6.3.1.1 WARM Start (Normal Restart)

The WARM start option initializes CA-7 with information in the queues remaining intact from the last time CA-7 was active. It may be used to reactivate CA-7 after a successful normal shutdown. The CA-7 internal control blocks are reinitialized from information residing in the checkpoint data set, communications data set, log data sets, and the queues.

All queues remain intact and secondary messages queued for terminals are retained. The queues and database are updated with any information stored in the communications data set while CA-7 was inactive.

The WARM start initialization process performs several checks of internal system status to ensure that a WARM start is, in fact, possible. If CA-7 detects that previously it did not shut down normally, it automatically performs an ERST start. It should be noted that this is not the same as a specified ERST type of startup. If TYPE=ERST is specifically coded, then queued messages are purged from the CA-7 queues. Also, if the number of terminals defined in the initialization file is greater than 254, then a TYPE=ERST is assumed. This is because CA-7 cannot checkpoint the terminal data in this case.

The following are considered errors during a WARM start:

- Any changes to the initialization file other than those specifically allowed. Allowed changes follow this list.
- Any discrepancy in the date, time and CA-7 version number information stored in the shutdown record in the checkpoint and log data sets.
- No shutdown record found in the log data set.

Changes to the initialization file are allowed as part of a WARM start. The allowable changes are:

Statement	Allowable changes
RESIDENT	Only the APGPL parameter may be changed.
INIT	All parameters may be changed.
SVCNO	The SASSVC parameter may be changed.
APPLCTN	Statements may be added or removed as necessary. All parameters may also be changed.
FMTBLK	Statements may be added or removed as necessary. All parameters may also be changed.
GROUP	Only the OPEN parameter may be changed.
LINE	Only BUFSIZE and OPEN parameters may be changed.
TERM	The RELADR, TIMLIM, MONLIM, LOGMODE, LCLSNA, ADRCHR, and CONS parameters only may be changed.

No GROUP, LINE, or TERM statements may be added or deleted. Sequence from previous initialization also must not change.

6.3.1.2 ERST Start (Emergency Restart)

The ERST start (emergency restart) option may be used to recover from most CA-7 system failures. It may also be required for certain initialization file changes. The object of emergency restart is to reconstruct the system as it existed prior to the system failure and to bring all status up to the current time.

During the ERST procedure terminals may be added, deleted, or changed. The security may be changed also.

Data stored in the communications data set (while CA-7 was down) is used to update the status. Upon successful emergency restart, the status of jobs under CA-7 is current as reflected in the queues; however, any messages queued to terminals are lost. All other information in the queues remains intact.

6.3.1.3 COLD Start (Clear Workload Restart)

The COLD start option is used when CA-7 queue information is not to be retained.

It is used when a nonrecoverable failure has occurred or whenever all information in the queues remaining from the last time CA-7 was active must be purged. It may also be required for certain initialization file changes. The environment definition is reestablished after every COLD start. The CA-7 queues are cleared with the possible exception of the prior-run and trailer queues. All internal control blocks, the log data sets, and the check-point data set are reinitialized. The workload balancing definition is reset to the default module UCC7RDFL, unless the DFLTWLB parameter of the OPTIONS statement in the initialization file is used. The schedule scan processing options are also reestablished and scanning may be initiated at the user's option. DISP=F on any format statement causes that particular queue to be formatted. Scanning may be initiated at the user's option.

When a COLD start is done, CA-7 starts logging to the primary log. Any data that was in either the primary or secondary log file is overwritten without being written to the history file. If the log data needs to be saved, this should be done by executing the log dump job(s) prior to doing the COLD start.

Note: When a COLD start is referred to in other CA-7 documentation, FORM and MOVQ could also be used. (FORM and MOVQ are considered COLD types of startups.)

The Virtual Resource Management facility (VRM) deletes all active, nonfreed, corequisite, resource count, and manual resource records during a CA-7 COLD or FORM start.

The following general considerations should be kept in mind:

- COLD start is necessary when recovery from an ERST error is not possible.
- A COLD start can optionally reformat queues by specifying DISP=F on the FORMAT statements in the initialization file for the queues to be reformatted.
- Internal CA-7 status, queue status and terminal status are initialized from scratch, regardless of the activity occurring when the previous online execution of CA-7 terminated.
- The default workload balancing module is always used for the initial values for WLB purposes.
- The primary log data set is always used for recording activity.

6.3.1.4 MOVQ Start (Move Queues Restart)

The MOVQ start is used to format and reload the queues prior to resuming activities. This is the recommended startup type for version upgrades. The MOVQ feature is identical to the FORM start in the initialization process except the CA-7 queues are reloaded from the queue dump data set. Once the queues are successfully reloaded, CA-7 continues as though a TYPE=COLD was entered. See the /SHUTDOWN command option DMPQ for more information on how to first unload the queues. (See A.1.1, “Unloading the Queues” on page A-3.)

Use the MOVQ option only after a successful shutdown has occurred with a DMPQ option.

A user exit (SASSXXQB) is provided to allow the user to look at, modify, delete or add records going to the queues. The MOVQ feature can be used only after a previous execution of CA-7 has been completed and a file created on //UCC7QDMP DD.

A //UCC7QDMP DD statement must be included in the UCC7 online JCL before entering the /SHUTDOWN,DMPQ. The file must be allocated as contiguous and large enough to hold the data from these files:

```
//UCC7PREQ DD  
//UCC7REQQ DD  
//UCC7RDYQ DD  
//UCC7ACTQ DD  
//UCC7PSTQ DD  
//UCC7PRNQ DD  
//UCC7TRLQ DD
```

Although SCRQ and DQTQ are not reloaded when using MOVQ, these files are reformatted.

6.3.1.5 FORM Start (Format Queues Restart)

FORM start must be used for the first execution of CA-7 following installation. It is also used whenever empty queues need to be reformatted (for example, after space reallocation or when they have been moved). After formatting, CA-7 continues as though a TYPE=COLD was entered. If queues have data that must be preserved even though they were reallocated or moved, see the MOVQ startup type and the /SHUTDOWN command DMPQ option. See 6.3.1.4, “MOVQ Start (Move Queues Restart)” on page 6-21.

Note: The Virtual Resource Management facility (VRM) deletes all active, nonfreed, corequisite, resource count, and manual resource records during a CA-7 COLD or FORM start.

6.3.1.6 DORM Start (Dormant CA-7)

DORM start allows a dormant (backup) copy of CA-7 to be started. This CA-7 will not fully initialize if the main CA-7 is active. However, it will monitor the status of the main CA-7 and if the main terminates, the dormant copy will complete initialization and take over for the main. See 6.4, “Dormant CA-7” on page 6-23.

6.3.1.7 EDIT Start (Offline Verification of Initialization File)

The EDIT start may be used to test values in the initialization file. TYPE=EDIT may be specified but does not cause CA-7 to be initialized. TYPE=EDIT is only valid on the PARM to CA-7 (that is, it is not a valid type in the initialization file). It is used only for initialization file editing. See 5.3, “Offline Verification of Initialization File” on page 5-86.

6.4 Dormant CA-7

CA-7 provides the capability of running a secondary, dormant, CA-7 that can be used to take over for a primary CA-7 which has terminated. This allows you to have an automatic recovery procedure for the CA-7 task in case of abnormal termination of the primary task or loss of the CPU where the primary was executing. The dormant CA-7 is started while the primary is active and monitors the status of the primary. If the dormant CA-7 determines that the primary has terminated, the dormant completes initialization and assumes the functions of the primary.

The dormant CA-7 can be executed on the same CPU as the primary or on another CPU that has access to the DASD where the CA-7 data sets reside.

6.4.1 Procedure

The dormant CA-7 must have access to all of the data sets that the primary CA-7 is using. To set up the JCL, you can copy the procedure for the primary CA-7 and simply change the TYPE= PARM. The dormant CA-7 executes under a different task/job name, so any security definitions that are in place for the primary CA-7 must be duplicated for the dormant copy.

A CA-7 is designated as dormant by specifying TYPE=DORM as the PARM in the JCL (see 6.3.1, "Startup Options" on page 6-18 for information about the Startup Options for CA-7). When this PARM is used, the CA-7 task starts and does some of the initialization process, and then issues the WTOR:

CA-7.740 - CA-7 DORMANT START. REPLY "YES" TO TAKE OVER OR "STOP" TO END

This WTOR allows you to stop the dormant CA-7 or manually request it to take over for the primary CA-7. At this point, the dormant CA-7 monitors the status of the primary CA-7 by checking the availability of the ENQ that is issued by CA-7 at startup. As long as the ENQ cannot be obtained, the dormant CA-7 remains in this state. When the ENQ can be obtained, the dormant CA-7 issues the following WTOR:

CA-7.743 - MAIN CA-7 INACTIVE - DORM WILL TAKE OVER UNLESS "STOP" REPLY

If a STOP response has not been received within 60 seconds (default), the dormant CA-7 completes initialization, ERST starts, and then takes over the functions from the primary CA-7. You can set the time interval using the DORMVER keyword of the SVCNO statement in the initialization file.

6.4.2 ENQ Monitoring

The monitoring of the primary CA-7 is accomplished by checking the availability of the ENQ issued by the primary. The default ENQ is:

	QNAME	RNAME
Production copy	UCC7SVT	UC07
Test copy	UCC7SVT	UCT7

Note: The RNAME may be overridden with the RNAME keyword on the SVCNO statement of the initialization file. The RNAME for the primary and dormant CA-7s must match. If they do not match, they can cause serious damage to the CA-7 database and queue files, requiring possible database reload and COLD start of CA-7.

If the dormant copy of CA-7 executes on a CPU different from the primary copy, you must ensure that an ENQ package is being used to enforce single usage of the above ENQ by any task on the CPUs.

6.5 Shutdown Procedures

The `/SHUTDOWN` command causes a normal termination of CA-7 execution. The command is only authorized from the CA-7 master or alternate terminal because of the impact of shutting down the CA-7 system. See 5.2.12, “TERM Statement” on page 5-33 for information on the `CONS` parameter.

For more information on the `/SHUTDOWN` command, see the *CA-7 Commands Guide*.

6.5.1 OS CANCEL Command

While not recommended, you can use the `OS CANCEL` command from the system master console to terminate CA-7 execution. You should use an `OS CANCEL` only when the `/SHUTDOWN` is unable to terminate CA-7. This command results in an abnormal termination, and the system issues an abend message. If the `OS CANCEL` is needed, specify the option to produce a dump to aid in problem analysis.

Terminals are not notified that the system is being deactivated. Consequently, some data may be lost.

After an `OS CANCEL` of CA-7, the `ERST` start option is the recommended method of reactivating CA-7.

6.6 ICOM Execution

ICOM is the Independent Communications Manager. It performs the functions described in the "System Structure" chapter of this manual. ICOM's main purposes are:

- Move SVC-captured SMF and trailer data to the communications data set for processing by CA-7 or to the CA-7 ICOM Coupling Facility list structure for processing by another ICOM.
- Optionally, to submit jobs to the host.

The ICOM task can also optionally be used to track Cross-Platform jobs and events. This is done by running the CA-7 Cross-Platform Tracking System (XTRK) as a subtask in the ICOM address space. CA-7 XTRK can also be executed as a separate job or started task. See Cross-Platform Tracking in Chapter 5 of the *CA-7 Interfaces Guide* for more information about the CA-7 Cross-Platform Tracking System. See 6.6.6, "Cross-Platform Tracking under ICOM" on page 6-35 for specific information on executing CA-7 XTRK under ICOM.

6.6.1 ICOM Initialization

ICOM must be activated on each CPU to perform automatic monitoring of CPU jobs and, optionally, job submission. ICOM, like CA-7, can be run as a batch job or a started task.

The way ICOM is initialized is determined by the PARM values on the EXEC statement. Following are an ICOM JCL example and descriptions of the PARMs. Then the general considerations for ICOM usage are discussed.

After initialization, you can change certain parameters and values passed to ICOM for the initialization by replying to ICOM's outstanding WTOR or by issuing a MODIFY command. The acceptable replies to the WTOR or MODIFY and their meanings are discussed later in this chapter.

We recommend that you read this topic before initializing ICOM. This is especially true of the user who has the choice of job submission by ICOM or CA-7.

```
//jobname JOB .....
//ICOM  PROC  SMFTRL=B,HOST=2,DD='**NONE**',R=256K,CF=,TEST=
//ICOM  EXEC  PGM=SASSICOM,REGION=&R,TIME=1440,
//        PARM='&HOST&DD&SMFTRL&CF&TEST'
//STEPLIB DD  DISP=SHR,DSN=ca7.loadlib
//SYSUDUMP DD  SYSOUT=A
//UCC7CMDS DD  DISP=SHR,DSN=ca7.commds
//UCC7WTR DD  SYSOUT=(A,INTRDR),DCB=BLKSIZE=80
//UCC7SUB1 DD  DISP=SHR,DSN=ca7.submtds1 (optional for shared spool)
//        PEND
//STEP1  EXEC  ICOM
```

Figure 6-5. ICOM Execution JCL

6.6.2 ICOM PARM Values

PARM data specified in the SASSICOM EXEC statement identifies the host system, the submit data set for that ICOM and the type of record collection to be done.

The following describes the usage of symbolic parameters from ICOM Execution JCL on page 6-26.

Where:

HOST

Identifies the host system and must be one of the following:

2

JES2

3

JES3

DD

Identifies the submit data set for this ICOM. Value must be one of the following:

UCC7SUBn

Name of the DD statement defining the submit data set.

NONE

No submit data set is defined. Job submission is not done by ICOM. CA-7 is to submit the jobs directly to an internal reader. (See 5.2.22, "CPU Statement" on page 5-54.)

SMFTRL

Specifies the type of data to be collected. Value must be one of the following:

B

Both SMF and trailer step data is collected. Value should be B for normal execution of ICOM.

N

No SMF or trailer step data is collected.

S

Only SMF data is collected.

T

Only trailer step data is collected.

TEST

Indicates use of a test copy of CA-7. If used, value must be TEST='TEST=YES'.

CF

Indicates the ICOM Coupling Facility Mode to be used. This parameter is optional. If used it should be in the form CF=x where x is one of the values listed below.

N

No Coupling Facility interface (this is the default).

W

Write to Coupling Facility. SMF, Trailer and NCF data collected by this ICOM will be written to the CA-7 ICOM Coupling Facility list structure.

R

Read from Coupling Facility. Besides collecting local SMF, Trailer and NCF data, this ICOM will also read data from the CA-7 ICOM Coupling Facility list structure. All data will be written to the CA-7 Communications data set (or NCF Communications data set).

Table 6-2. ICOM DD Statements	
DDNAME	Explanation
STEPLIB	Defines the CA-7 load module library.
SYSUDUMP	Defines a standard SYSUDUMP output data set.
UCC7CMDS	Defines the communications data set. It is always required. Must be the same data set name as in the CA-7 execution JCL.
UCC7WTR	Defines the internal reader to which JCL submitted by ICOM is written. The SYSOUT=INTRDR provides a symbolic reference to an internal reader in JES2 and JES3 environments. The UCC7WTR DD is required whenever ICOM is to perform job submission. If no job submission is to be done by ICOM (DD PARM of **NONE**), it should be omitted.
UCC7SUBn	Defines the submit data set to be referenced by the ICOM. The UCC7SUBn name must match the corresponding ddname in the CA-7 initialization file (CPU statement DDNAME) and the DD parameter of the SASSICOM PARM value. The UCC7SUBn DD is required whenever ICOM is to perform job submission. If no job submission is to be done by this ICOM (the PARM data DD parameter specifies **NONE**), it may be omitted.

6.6.3 General Usage Considerations for ICOM

General considerations for the use of ICOM JCL, initialization, and execution of ICOM are:

- The ddnames in ICOM JCL are fixed except for the variable portion indicated in the ddname of the statement defining the submit data set.
- If submit data sets are used, the data set name (in the DD statement specified by the DD parameter) must be the same data set name as that coded in the CA-7 execution JCL (pointed to by the DDNAME keyword in the CPU initialization file statement). Each ICOM has its own submit data set. The CA-7 JCL must have a corresponding submit data set, pointed to by a CPU initialization file statement (that is, there is a CPU statement for each ICOM). Again, this is only true if submit data sets are used instead of the internal reader option discussed below.
- In a shared spool environment, the submit data set can be omitted. (See 5.2.22, “CPU Statement” on page 5-54.) In this case, the DD parameter for ICOM specifies ****NONE****.

6.6.4 ICOM WTOR/MODIFY Replies

The Independent Communications Manager (ICOM) must be initialized and active for CA-7 to perform automatic job submission and monitoring of CPU job activity through SMF.

When ICOM initialization is successfully completed, a WTOR is issued to the operating system master console. The format of this WTOR is:

```
CA-7.574  ENTER ICOM REQUEST
```

This message remains as an outstanding WTOR as long as ICOM is active. It may be used to alter or terminate an ICOM operation or to inquire into its current state of activity. However, no reply is required for ICOM to function as specified by its initialization JCL parameters. Alternatively, if ICOM is a started task, a MODIFY command may be used instead of the WTOR reply.

Replies to ICOMs WTOR (CA-7.574) or MODIFY input may be used to perform the following functions:

- Display ICOM control blocks
- Dynamically alter ICOM processing options
- Shut down ICOM

WTO messages are issued to the (OS) master console indicating the results of a particular reply to ICOM's WTOR.

The possible replies to the ICOM WTOR and the MODIFY command are discussed in the remainder of this chapter. WTOs which result from one of these replies are also explained. WTOs which are not a direct result of a reply to the ICOM WTOR or MODIFY are explained in the *CA-7 Message Guide*.

The replies are:

STOP Used to terminate ICOM processing. WTOs are issued confirming the tasks were stopped. No new outstanding CA-7.574 WTOR is issued. (In general, an OS CANCEL should not be issued for ICOM.)

CF=STAT

Used to display the status of the ICOM Coupling Facility interface and statistics for the ICOM list structure.

```
CA-7.586 ICOM COUPLING FACILITY MODE = xxxxxx
```

Where xxxxxxxx is NONE, READ, or WRITE.

```
CA-7.587 ICOM CF : CURRENT   : SMF sssssss TRL ttttttt NCF nnnnnn
CA-7.587 ICOM CF : PROCESSED : SMF sssssss TRL ttttttt NCF nnnnnn
```

Where:

sssssss Is the number of current/processed SMF records

ttttttt Is the number of current/processed Trailer records

nnnnnnn Is the number of current/processed NCF records

See the *CA-7 Message Guide* for a detailed explanation of these messages.

D=xxx...x Used to display memory at the location indicated by xxx...x, where xxx...x may be one of the following:

DSECT Dump the first X'80' bytes of ICMDSECT (which is memory resident). A CA-7.576 WTO is issued to display this. A new CA-7.574 WTOR is issued following the response.

ICOM Dump the first X'20' bytes of ICOM. A CA-7.576 WTO is issued to display this. A new CA-7.574 WTOR is issued following the response.

STAT Display statistics for SMF and trailer as well as the ddname of the submit data set. A CA-7.575 WTO is issued to display the information. A new outstanding CA-7.574 is issued following the response.

WTO responses to D=STAT are in the following format:

```
CA-7.575 SMF - aaa(bbb) ccccccc(dddddddd) xxx
CA-7.575 TRL - aaa(bbb) ccccccc(dddddddd) xxx
CA-7.575 DDNAME=xxx...x
```

See the *CA-7 Message Guide* for a detailed explanation of each field.

SVC Dump the first X'40' bytes of the first load of the CA-7 SVC. A CA-7.576 WTO is issued. A new CA-7.574 WTOR is issued following the response.

EXTJOB Display entries in the SASSEXTL table currently being used. A new outstanding CA-7.574 is issued following the response.

EXTDSN Display entries in the SASSXDSN table currently being used. Message CA-7.585 is issued for each entry in the table. A new outstanding CA-7.574 is issued following the response.

DD=ddname

Used to indicate the ddname of a submit data set to be used for job submission. The ddname must be defined in ICOM's JCL. This reply may be required if multiple submit data sets are defined in the ICOM JCL and it becomes necessary to switch from one to another. If the ddname is being changed from ****none****, the S=SUB command may be used to start job submission.

No WTO response is issued. A new outstanding CA-7.574 WTOR is issued.

DEL=xxxxx

This is a problem debugging aid and should only be used with the guidance of a CA-7 specialist. Used to delete chains of SMF or trailer records from the CSA/ECSA area.

No WTO response is issued. A new outstanding CA-7.574 WTOR is issued.

P=xxx...x

Used to stop an ICOM function which is currently active when the reply is entered. The function designated by xxx...x, may be any of the values allowable for S=xxx...x (below), which is the reply used to start inactive functions of ICOM.

No WTO response is issued. A new outstanding CA-7.574 WTOR is issued.

S=xxx...x

Used to start an ICOM function not currently active when the reply is entered. The function, designated by xxx...x, may be any one of the following:

CALL Start collecting both SMF and trailer records.

CSMF Start collecting only SMF records.

CTRL Start collecting only trailer records.

FILBUF Require buffers to be filled before the buffers are written to the communications data set. (If this is not specified, buffers are output to the communications data set by the occurrence of a job initiation, step termination, job termination, or JCL error.)

IOALL Start writing both SMF and trailer records to the communications data set.

IOSMF Start writing only SMF records to the communications data set.

IOTRL Start writing only trailer records to the communications data set.

SUB Start the submit function.

No WTO response is issued. A new outstanding CA-7.574 WTOR is issued.

SUB=x SUB=x is a problem debugging aid and should only be used with the guidance of a CA-7 specialist. Used to modify the submission process for problem debugging. The value specified for x must be either F to force submission of jobs in the submit data set even if CA-7 is inactive, or R to cause resetting of control bits in the communications data set. No WTO response is issued. A new outstanding CA-7.574 WTOR is issued. (Not valid unless submit data sets are used.)

6.6.5 ICOM Coupling Facility Exploitation

The Coupling Facility is a memory extension or common storage for each processor in a Sysplex environment. The Coupling Facility can be used as a shared I/O device for all participating processors.

Without the Coupling Facility, each ICOM must write feedback data directly to the communications data set. With the Coupling Facility a "memory" list can be shared by multiple ICOMs. Most ICOMs will write feedback data into the Coupling Facility list (WRITE mode) while one or more ICOMs will read feedback data from the Coupling Facility list (READ mode) and write it to the communications data set. Such a configuration significantly reduces physical contention for the communications data set thereby increasing feedback throughput.

There are six lists within the ICOM Coupling Facility structure:

1. CA-7 SMF data for the production (primary) copy of CA-7.
2. CA-7 Trailer data for the production (primary) copy of CA-7.
3. CA-7 NCF data for the production (primary) copy of CA-7.
4. CA-7 SMF data for the test (secondary) copy of CA-7.
5. CA-7 Trailer data for the test (secondary) copy of CA-7.
6. The sixth list is not used at this time.

Each list within the ICOM Coupling Facility structure is treated as a FIFO queue (first-in first-out). New entries are added to the end of the list while existing entries are read from the top of the list. List data elements are 4k (4096 bytes) in size.

For specific information on defining an ICOM Coupling Facility, see 3.3, "ICOM Coupling Facility Structure" on page 3-27.

6.6.5.1 Coupling Facility Operating Modes for ICOM

1. WRITE Mode

WRITE mode causes ICOM to write the SMF, Trailer, and NCF data it collects to the ICOM Coupling Facility structure. Data elements are added to the end of the appropriate list in the structure.

2. READ Mode

READ mode causes ICOM to collect SMF, Trailer and NCF data from the local environment. Also, ICOM reads data from the ICOM Coupling Facility structure. All data collected is written to the communications data set or the NCF communications data set. Coupling Facility entries are read from the top of each list using a read/delete function.

6.6.6 Cross-Platform Tracking under ICOM

Cross-Platform tracking consists of the XPS SERVER returning feedback records for CA-7 submitted jobs to the system which sent the request. These records are transmitted using CAICCI and represent job initiation, job termination, and job failure events.

The CA-7 Cross-Platform Tracking System (XTRK) running as a started task, batch job, or as a subtask under ICOM on the OS/390 system where the CA-7 Cross-Platform Submit job executed receives these events and converts them into SMF like feedback recognized by CA-7. CA-7 then processes this information like any other job to determine when it becomes active, completes, or fails.

Also, events such as job execution and file creation/update occurring on non-OS/390 platforms can be tracked even though CA-7 did not request that work (Cross-Platform External Tracking). Unicenter TNG or a CA-7 Agent must be executing at the remote node for this tracking to occur.

One copy of CA-7 XTRK should be running on each system where CA-7 Cross Platform Submit (CA7TOUNI) jobs can execute. CA-7 ICOM must also be executing on these systems.

See Cross-Platform Tracking in Chapter 5 of the *CA-7 Interfaces Guide* for more information about the CA-7 Cross-Platform Tracking System.

6.6.6.1 Cross-Platform Tracking ICOM PARM Values

If you wish to execute the CA-7 Cross-Platform Tracking System (XTRK) as a subtask in the ICOM address space, add the following parameters to the SASSICOM EXEC statement.

`XMONITOR=monitor-name`

This required parameter is the 7 character monitor name which uniquely identifies each copy of CA-7 whose cross-platform jobs are to be tracked. It must match the MONITOR= value used by the CA-7 Cross-Platform Submit jobs to be tracked (CA7TOUNI).

Note: This parameter is required if you wish to execute the CA-7 XTRK system in the ICOM address space. If this parameter is specified then the XCKPT DD statement for the CA-7 Cross-Platform Tracking Checkpoint file must be present in the ICOM JCL.

`XTRC=trace-codes`

These optional codes specify the level of diagnostic messages and snaps that should be generated by XTRK. There are two codes that can be specified: print/snap trace code and console message trace code. The first value controls what messages will be written to the XPRINT DD, and what records and control blocks will be written to the XSNAP DD. The second value controls what WTOs are issued to the system console.

The default trace code is 'XTRC=11'. See Cross-Platform Tracking JCL in the *CA-7 Interfaces Guide* for detailed descriptions of the trace code values.

Note: If used, the XMONITOR= and XTRC= keywords should be added to the end of the SASSICOM parameter list and preceded by commas to separate them from other PARM values.

6.6.6.2 Cross-Platform Tracking ICOM DD Statements

If you wish to execute the CA-7 Cross-Platform Tracking System (XTRK) as a sub-task in the ICOM address space the following DD statements should be added to the SASSICOM JCL.

XCKPT

The XCKPT DD must point to a CA-7 Cross-Platform checkpoint file. Each copy of XTRK must have its own checkpoint file. That is, if you are running ICOM on two separate MVS images and you wish to run CA-7 XTRK, each copy must have a separate checkpoint file. See Cross-Platform Tracking JCL in the *CA-7 Interfaces Guide* for detailed specifications for CA-7 Cross-Platform Checkpoint files.

Note: The XCKPT DD statement is required if the XMONITOR= keyword is specified in the SASSICOM EXEC parameters.

XEVENTS

The XEVENTS DD is optional. If specified, it should point to an 80-byte card-image sequential data set, or PDS member, which contains CA-7 cross-platform external tracking rules. These rules define what events that occur on other systems should be tracked by CA-7 even though CA-7 did not submit the work that caused these events to take place. See Cross-Platform External Tracking in Chapter 5 of the *CA-7 Interfaces Guide* for information on the format of these rules.

Note: If you are running CA-7 XTRK on multiple MVS images, you should NOT specify the same external tracking rules for more than one copy of CA-7 XTRK. This could cause CA-7 to receive multiple copies of the same tracking information from different copies of CA-7 XTRK.

XPRINT

If specified, the XPRINT DD should specify a SYSOUT class for CA-7 Cross-Platform Tracker (XTRK) trace messages. The type and volume of messages produced is controlled by the first value of the XTRC= keyword in the SASSICOM EXEC parameters.

XSNAP

If specified, the XSNAP DD should specify a SYSOUT class for CA-7 Cross-Platform Tracker (XTRK) storage snap output. The type and volume of output produced is controlled by the first value of the XTRC= keyword in the SASSICOM EXEC parameters.

6.6.6.3 Cross-Platform Tracking ICOM WTOR/MODIFY Replies

If you execute the CA-7 Cross-Platform Tracking System (XTRK) as a subtask of ICOM you can pass commands to the XTRK task using the CA-7.575 WTOR or a MODIFY command. Responses to these commands are issued as WTOs and are explained in the *CA-7 Message Guide*.

The ICOM reply/command for XTRK is:

XTRK=...native XTRK command...

For example, to issue a MODIFY command to display all nodes connected to XTRK issue:

F CA7ICOM,XTRK=NODE

See Cross-Platform Tracking Commands in Chapter 5 of the *CA-7 Interfaces Guide* for detailed specifications of specific XTRK commands.

6.7 CAIRIM Initialization Considerations

6.7.1 Overview

The system interfaces for CA-7 are initialized using the Computer Associates Resource Initialization Manager, CAIRIM. CAIRIM is a Unicenter TNG Framework for OS/390 component (S910). CAIRIM is the common driver for a collection of dynamic initialization routines that eliminate the need for preinstalled user SVCs, SMF exits, subsystems, and other installation requirements commonly encountered when installing systems software. See Unicenter TNG Framework for OS/390 *Getting Started* for a full discussion of CAIRIM requirements and execution.

6.7.2 CAIRIM Execution

CAIRIM should be run during or immediately after each IPL to initialize the CA-7 system interfaces. Other CA products may also be initialized during the same execution of CAIRIM. If this is not done, CA-7 still executes, but it cannot track jobs. ICOM abends with a user code 20 during startup if the CA-7 system interfaces have not been initialized.

CAIRIM should only need to be run once during the life of an IPL. However, should you need to run it again for another CA product, CA-7 is not affected. You receive messages indicating CA-7 is already initialized so no action is taken.

Note: If SMF processing abends and has to be restarted (for example, using SET SMF console command), then CAIRIM must be rerun to establish its interface.

6.7.3 CA-7 Reinitialization Using CAIRIM

CAIRIM can be used in special situations to reinitialize the CA-7 system interfaces without an IPL. This may be required for CA-7 maintenance and version upgrades. The advantage to reinitializing the CA-7 interfaces with CAIRIM is that an IPL is not required. The disadvantage is that some CSA/ECSA storage is lost until the next IPL, and in some cases, tracking for jobs in progress may be lost.

CA-7 has three types of reinitializations: Normal, COLD, and User Table Reload.

6.7.3.1 Normal Reinitialization

During a Normal reinitialization, most of the CA-7 system interface modules are reloaded and the CA-7 system intercepts are refreshed. Most of the old modules are left in CSA/ECSA, so that you lose about 12K of CSA/ECSA storage for each reinitialization which is not recovered until the next IPL.

The only modules which are not refreshed during a normal reinitialization are ICMDSECT and UCC7NODE (a CA-7 NCF control table). Module ICMDSECT is not refreshed because CA-7 trailer and SMF records in CSA/ECSA are chained off of this module. If it were replaced, CA-7 would lose the tracking information in those records. The UCC7NODE module is also chained off of the ICMDSECT module, so a normal reinitialization cannot be used to refresh it.

6.7.3.2 COLD Reinitialization

During a COLD reinitialization, modules ICMDSECT and UCC7NODE are refreshed, in addition to the other modules and interfaces which are affected by a normal reinitialization. The ICMDSECT initialization process performed by the CA-7 SVC module is repeated for the new copy of ICMDSECT. Additionally, the SSCT entry for CA-7 (UC07) is upgraded to point to the new ICMDSECT module. Some of the old modules and buffers are left in CSA/ECSA, so that you lose at least 24K of CSA/ECSA storage for each COLD reinitialization which is not recovered until the next IPL.

Note: When a cold reinitialization is performed, any SMF and trailer records chained off of the old ICMDSECT are lost. This may cause tracking for some jobs to be missing or incomplete.

After a COLD reinitialization, ICOM must be shut down and restarted so that it addresses the new copy of ICMDSECT in CSA/ECSA. If you are running CA-7 NCF, it also has to be shut down and restarted.

If you attempt a COLD reinitialization before CA-7 has already been initialized, it is treated as an error, and CA-7 is not initialized at all.

6.7.3.3 User Table Reload Reinitialization

During a User Table Reload reinitialization, only the user-supplied table modules residing in CSA/ECSA are refreshed. These tables control external job and data set tracking, and data set tracking inclusion/exclusion criteria. The tables which are refreshed are SASSEXTL, SASSXDSN, and SASSXU83. Some CSA/ECSA storage for these tables is lost the first time any type of CA-7 CAIRIM reinitialization is performed. Subsequent User Table Reload reinitializations do not cause the loss of any additional CSA/ECSA.

6.7.4 Reinitialization Procedures

To perform a CA-7 reinitialization using CAIRIM, you need to set up a special CAIRIM initialization file in the CA Options file. This file is used only for reinitializations of CA-7. A new keyword is added to the CA-7 initialization statement, PARM(XXXXX), which causes reinitialization to occur. Never add reinitialization PARMs to your production CAIRIM initialization file.

Note: See Unicenter TNG Framework for OS/390 *Getting Started* for information on CAIRIM initialization files and executing the CAIRIM procedure.

6.7.4.1 Normal Reinitialization Procedures

Step	Action
1	<p>Create a new member (L233RINT) in the CA options file. Copy your existing CAIRIM initialization file (default name is CARIMPRM) into the new member and <u>delete all statements except the one for CA-7</u>. On the CA-7 statement, add PARM(REINIT) to the end of the statement. An example of the updated statement is:</p> <pre>PRODUCT(CA-7) VERSION(L233) PARM(REINIT)</pre>
2	<p>Execute the CAIRIM started task using the L233RINT file:</p> <pre>START CAS9,RIMPARM=L233RINT</pre> <p>You receive the following message in addition to the normal CA-7 initialization messages:</p> <pre>CAL2004I CA-7 RIM: REINIT PARM ACCEPTED. CA-7 WILL BE REINITIALIZED.</pre>

6.7.4.2 COLD Reinitialization Procedures

Step	Action
1	<p>Create a new member (L233COLD) in the CA options file. Copy your existing CAIRIM initialization file (default name is CARIMPRM) into the new member and <u>delete all statements except the one for CA-7</u>. On the CA-7 statement, add PARM(REINIT=COLD) to the end of the statement. An example of the updated statement is:</p> <pre>PRODUCT(CA-7) VERSION(L233) PARM(REINIT=COLD)</pre> <p>Note: The old syntax of PARM(REINIT(COLD)) is still honored. However, if combined with other PARMs, it must be the last entry. For example, PARMS(...other parms...,REINIT(COLD)).</p>
2	<p>Execute the CAIRIM started task using the L233COLD file:</p> <pre>START CAS9,RIMPARM=L233COLD</pre> <p>You receive the following messages in addition to the normal CA-7 initialization messages:</p> <pre>CAL2004I CA-7 RIM: REINIT PARM ACCEPTED. CA-7 WILL BE REINITIALIZED. CAL2005I CA-7 RIM: ***** ATTN: CA-7 COLD REINIT IN PROGRESS.</pre>
3	<p>After the COLD reinitialization is complete, shut down and then restart ICOM. You must do this so that ICOM can connect with the new copy of ICMDSECT in CSA/ECSA where the tracking records are now being chained. If you are running CA-7 NCF, it should also be shut down and restarted.</p>

6.7.4.3 User Table Reload Reinitialization Procedures

Step	Action
1	<p>Create a new member (L233UTAB) in the CA options file. Copy your existing CAIRIM initialization file (default name is CARIMPRM) into the new member and <u>delete all statements except the one for CA-7</u>. On the CA-7 statement, add PARM(REINIT=UTABS) to the end of the statement. An example of the updated statement is:</p> <pre>PRODUCT(CA-7) VERSION(L233) PARM(REINIT=UTABS)</pre>
2	<p>Execute the CAIRIM started task using the L233UTAB file:</p> <pre>START CAS9,RIMPARM=L233UTAB</pre> <p>You receive the following message:</p> <pre>CAL2038I CA-7 RIM: USER TABLE RELOAD INITIALIZATION PROCESS COMPLETE.</pre>
3	<p>You do NOT have to recycle CA-7 or ICOM after a User Table Reload initialization. The new copies of the user tables are used as soon as the reinitialization process is complete.</p>

6.8 Other CAIRIM Initialization Parameters

The following parameters can also be specified in the PARM field of the CAIRIM initialization statement for CA-7. Unless otherwise noted they can be used in combination with other CA-7 CAIRIM initialization parameters (such as REINIT).

6.8.1 Specifying the Host NCF Node

If you are using the CA-7 Network Communications (NCF) facility you can specify the NCF=xx parameter on the CAIRIM initialization statement for CA-7 to designate the node which should be considered the host (that is, local) node for the system being initialized. The value (xx) should match the CA-7 identifier for the host entry in the NCF node table (UCC7ID= value on the UNCNOD entry for the host system). This parameter is optional. If it is not coded and the current system SMF ID does not match an entry in the table, the first entry in the table will default as the host node.

If the CA-7 identifier for the system being initialized is 78, the CA-7 initialization statement would be:

```
PRODUCT(CA-7) VERSION(L233) PARM(NCF=78)
```

If you wish to use the parm with a normal reinitialization PARM, the format would be:

```
PRODUCT(CA-7) VERSION(L233) PARM(NCF=78,REINIT=COLD)
```

Note: The old syntax of PARM(NCF(xx)) is still honored. However, if combined with other PARMS, it must be the last entry. For example, **PARMS(...other parms...,NCF(xx))**.

6.8.2 Forcing the CA-7 SVC

Before CAIRIM dynamically installs the CA-7 SVC on the system a check is made to determine if that SVC is already in use. If so message CAL2029W is issued and CA-7 will not be initialized. If you wish to force CAIRIM to use that SVC number despite this, specify the FORCESVC parameter on the CAIRIM initialization statement for CA-7.

The format of the statement would be:

```
PRODUCT(CA-7) VERSION(L233) PARM(FORCESVC)
```

The FORCESVC parameter can be used with other CA-7 CAIRIM initialization parameters.

6.8.3 Clearing the CA-7 CSA/ECSA Chains

If a problem occurs with the chaining of CA-7 SMF feedback and trailer records you may be directed by CA-7 Technical Support to make a special run of CAIRIM to clear the CA-7 CSA/ECSA chains.

The format of the statement would be:

```
PRODUCT(CA-7) VERSION(L233) PARM(CHAINS)
```

The CHAINS parameter MUST BE THE ONLY PARAMETER SPECIFIED.

Note: The CHAINS parameter causes any SMF feedback or trailer information currently chained in CSA/ECSA to be lost. IT SHOULD BE USED ONLY AT THE DIRECTION OF CA-7 TECHNICAL SUPPORT.

Chapter 7. Backup and Recovery Considerations

This chapter discusses backup and recovery considerations for the CA-7 database, CA-7 terminals, and the CA-7 workload.

7.1 Database

Because CA-7 is controlling a production environment, backup and recovery of its database becomes extremely important. Backups of the CA-7 database should be scheduled on a regular basis, at least once each day. If possible, CA-7 should be down or at least reasonably inactive during the backup, with no permanent updates being made to the database. All data sets in the database must be backed up at the same time.

Additionally, the backup procedure should be as fast as possible especially if scheduling is to stop. Two other concerns for backups are to produce a single source for recovery and, where practical, to provide error checking of index and pointer elements.

With the above items in mind, you may find that no single utility satisfies all your concerns. On the one hand, the SASSBK00 program provided with CA-7 creates a single source file for recovery and performs error checking of index and pointer elements; however, it is slow for a large database. (It is slow because it creates a logical as well as a physical backup for conversion purposes and therefore produces many more records than a utility such as IDCAMS or CA-ASM2.) On the other hand, utilities such as CA-ASM2, IDCAMS, and DFDSS are fast and can produce a single source for recovery, but they have no error checking of elements.

7.1.1 Backup Procedures

SASSBK00, provided with CA-7, is the recommended utility for backups, but if it proves to be too slow for your installation, consider using two procedures. The first procedure would use a fast utility on a daily schedule. The second procedure could use SASSBK00 on a weekly or monthly schedule to provide error checking. Another alternative could be to use both utilities on a daily schedule but have the fast utility make a disk copy which SASSBK00 could process without interfering with production scheduling.

As for which fast (or alternative) utility can be used, any utility which can process your VSAM database works. The following descriptions reflect some common utilities but is by no means all of the utilities available.

7.1.1.1 IDCAMS

The following JCL illustrates using IDCAMS to put a database on one tape volume or set of volumes, depending on size. If you wish such a job to run only if CA-7 is down, you must use INFILE instead of INDATASET and specify DISP=OLD on the DD statement referenced by INFILE.

```
//jobname JOB .....
//*
//IDCAMS EXEC PGM=IDCAMS
//SYSIN DD *
  REPRO OUTFILE(FILE1) INDATASET(your.IDS.name)
  REPRO OUTFILE(FILE2) INDATASET(your.SASDS.name)
  REPRO OUTFILE(FILE3) INDATASET(your.SASJOB.name)
//SYSPRINT DD SYSOUT=*
//FILE1 DD DISP=(NEW,CATLG),
//        DCB=(RECFM=VB,LRECL=4096,BLKSIZE=32760),
//        DSN=backup.IDS.name, <== You may want to use a GDG
//        UNIT=tape, <== Specify desired unit type
//        VOL=(,RETAIN),
//        LABEL=1
//FILE2 DD DISP=(NEW,CATLG),
//        DCB=*.FILE1,
//        DSN=backup.SASDS.name, <== You may want to use a GDG
//        UNIT=AFF=FILE1,
//        VOL=(,RETAIN,REF=*.FILE1),
//        LABEL=2
//FILE3 DD DISP=(NEW,CATLG),
//        DCB=*.FILE1,
//        DSN=backup.SASJOB.name, <== You may want to use a GDG
//        UNIT=AFF=FILE1,
//        VOL=REF=*.FILE1,
//        LABEL=3
```

Note: This example should be checked thoroughly to ensure that it works in your specific environment.

7.1.1.2 CA-ASM2

A product which performs backups well and handles either database format easily is CA-ASM2. The following shows sample JCL and control statements to cause CA-ASM2 to back up the CA-7 database. This is a two-step process which first identifies the data sets with a unique queue name, then requests the queue to be processed. For more information on this process and additional options available, see the CA-ASM2 documentation. The JCL procedures referenced probably have different names at your installation.

```
//jobname JOB .....  
//*  
//QUEUEUP EXEC ASM2CMDU  
//SYSIN DD *  
$BK DSNAMES('your.IDS.name')  
$BK DSNAMES('your.SASDS.name')  
$BK DSNAMES('your.SASJOB.name')  
//*  
//BACKUP EXEC ASM2EXPB  
//UNLOAD30.SYSIN DD *  
$ARCHEXT  
$USRPASS  
/*
```

Note: This example should be checked thoroughly to ensure that it works in your specific environment.

7.1.1.3 DFDSS

Another product which handles backups for either database format is DFDSS (Data Facility Data Set Services) from IBM. The following is an example for performing a backup using DFDSS.

```
//jobname JOB .....
//*
//BACKUP EXEC PGM=ADRDSSU
//TAPE DD DISP=(NEW,CATLG),
//      DSN=your.backup.name, <== Could be GDG
//      UNIT=tape <== Specify your unit type
//DASD DD DISP=SHR,
//      UNIT=disk, <== Specify your unit type
//      VOL=SER=xxxxxx <== Volume containing database
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
      DUMP INDDNAME(DASD) OUTDDNAME(TAPE) -
          DATASET(INCLUDE(your.IDS.name, -
                          your.SASDS.name, -
                          your.SASJOB.name))
//*
```

Note: This example should be checked thoroughly to ensure that it works in your specific environment.

7.1.1.4 SASSBK00

All database data sets can be backed up into one sequential data set, using the CA-7 program SASSBK00.

Following is an example of the JCL required to back up a database using SASSBK00. In the example, the LIST and XREF options cause detailed inventory and cross-reference reports to be produced. A control report, SASSBK00-01, is produced automatically for each execution of SASSBK00. Additionally, the example shows the use of a user exit. Discussions of the PARM values are included later in 7.1.1.6, “SASSBK00 PARM Values” on page 7-9.

```
//jobname JOB .....
//stepname EXEC PGM=SASSBK00,PARM=('OPT=(BACKUP,LIST,XREF)',
//          'DB=ALLVSAM','EXITMOD=userprog','XREFMOD=SASSXREF')
//STEPLIB DD DSN=ca7.loadlib,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//SYSOUT DD SYSOUT=A
//BACKUP DD DSN=user.dsn,DISP=(NEW,CATLG,DELETE),UNIT=tape
//DBCTLRPT DD SYSOUT=A
//DBLIST DD SYSOUT=A
//DBXREF DD SYSOUT=A
//PDSWORK DD UNIT=disk,SPACE=(CYL,(5,5))
//XREFWORK DD UNIT=disk,SPACE=(CYL,(5,5))
//          DCB=(RECFM=FB,LRECL=22,BLKSIZE=nnnnn)
//RPTFILE DD UNIT=disk,SPACE=(CYL,(5,5)),DISP=(NEW,PASS),
//          DCB=(RECFM=FB,LRECL=100,BLKSIZE=nnnnn)
//SRTFILE1 DD UNIT=disk,SPACE=(CYL,(5,5)),DISP=(NEW,PASS),
//          DCB=*.RPTFILE
//SORTIN DD DSN=*.RPTFILE,DCB=*.RPTFILE,DISP=OLD,
//          UNIT=disk,VOL=REF=*.RPTFILE
//SORTOUT DD DSN=*.SRTFILE1,DCB=*.RPTFILE,DISP=OLD,
//          UNIT=disk,VOL=REF=*.SRTFILE1
//SORTWK01 DD UNIT=disk,SPACE=(CYL,10,,CONTIG)
//SORTWK02 DD UNIT=disk,SPACE=(CYL,10,,CONTIG)
//SORTWK03 DD UNIT=disk,SPACE=(CYL,10,,CONTIG)
//SORTWK04 DD UNIT=disk,SPACE=(CYL,10,,CONTIG)
//SORTWK05 DD UNIT=disk,SPACE=(CYL,10,,CONTIG)
//SORTWK06 DD UNIT=disk,SPACE=(CYL,10,,CONTIG)
//UCC7JLIB DD DSN=user.defined.job.dataset,DISP=OLD
//UCC7DLIB DD DSN=user.defined.dataset.dataset,DISP=OLD
//UCC7IDS DD DSN=user.defined.index.dataset,DISP=OLD
//DBPARMS DD DSN=allvsam.database.parms,DISP=SHR
```

Figure 7-1. Sample JCL for Database Backup

Note: To perform a reload, specify 'OPT=(RELOAD...)' on the EXEC statement. Change DISP on BACKUP DD to OLD.

This example should be checked thoroughly to ensure that it works in your specific environment.

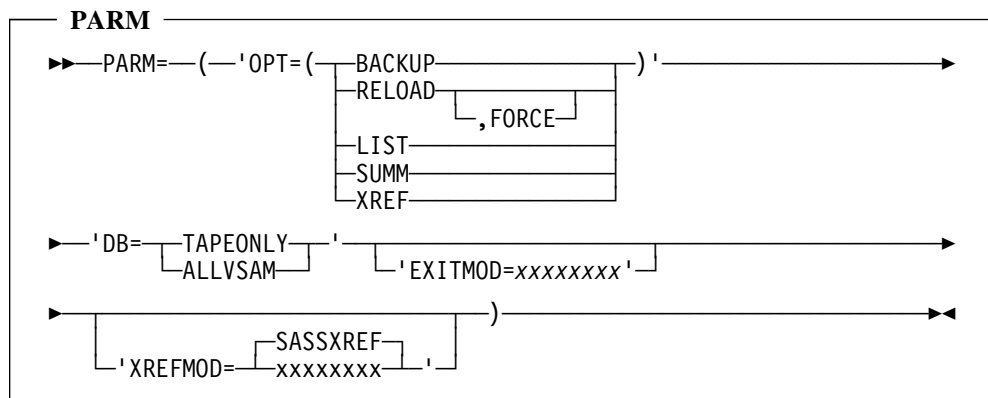
7.1.1.5 SASSBK00 DD Statements

STEPLIB	Must point to the load module library containing the CA-7 modules. If user exit modules are being used, as referenced by the EXITMOD and XREFMOD parameters, any user load module library containing those modules should be concatenated.
SYSUDUMP	Standard use and meaning.
BACKUP	<p>Points to the backup data set (GDG recommended), normally residing on tape. The BACKUP DD is always required. If the cross-reference report is requested without the BACKUP or RELOAD options and without DB=TAPEONLY, it should point to a temporary data set. If DB=TAPEONLY or OPT=BACKUP or RELOAD, it must point to the backup data set. Any user-defined BLKSIZE value may be used, up to the 32K limit imposed by the access method. By using a larger BLKSIZE value, the amount of tape or DASD required to hold the data is decreased. If the user does not specify the DCB, default DCBs are assigned as follows:</p> <ol style="list-style-type: none"> 1. RECFM=VB 2. LRECL=4096 3. BLKSIZE: <ol style="list-style-type: none"> a. TAPE=32760 b. 3390=27998 c. 3380=23476 d. 3375=17600 e. 3350=19069 f. OTHER=13030
DBCTLRPT	Used to produce control report SASSBK00-01. (See CA-7 DB Backup/Reload Control Report on page 7-20.) The report is automatically produced for each execution of the SASSBK00 program.
DBLIST	Used to produce database inventory report SASSBK07-01. (See Field Descriptions - SASSBK07-01 on page 7-24.) Required.
DBXREF	Used to produce database cross-reference reports SASSXREF-01 and SASSXREF-02. Required if XREF is specified.
SYSOUT	Standard use and meaning for SORT messages. Used only if XREF is specified.
RPTFILE	Temporary data set. Used only when XREF is specified.
XREFWORK	Temporary data set. Used only when XREF is specified.
SRTFILE1	Temporary data set. Used only when XREF is specified.
SORTIN	Sort input data. Used only when XREF is specified.
SORTOUT	Sort output data. Used only when XREF is specified.

SORTWKnn	Sort work areas used only when XREF is specified.
UCC7JLIB	Points to the job data set. If dynamic allocation (ALLOCDYN) is specified in DBPARMS, then the UCC7JLIB DD statement is <u>not</u> required and should be commented out.
UCC7DLIB	Points to the Dataset data set. If dynamic allocation (ALLOCDYN) is specified in DBPARMS, then the UCC7DLIB DD statement is <u>not</u> required and should be commented out.
UCC7IDS	Points to the index data set. If dynamic allocation (ALLOCDYN) is specified in DBPARMS, then the UCC7IDS DD statement is <u>not</u> required and should be commented out.
DBPARMS	Points to the UCC7DBASE control statements which further define the VSAM data sets. See 7.1.4, “UCC7DBASE Statements” on page 7-16 for coding instructions on these statements. The same values used in the UCC7DBASE statements when the database was last loaded must also be used when OPT=BACKUP is coded, to ensure correct access to the database.
PDSWORK	Optional temporary data set used for error checking. If omitted, checking is done in core which may require a large region size. It must reside on DASD and DCB values should not be specified since they are assigned by the program.

7.1.1.6 SASSBK00 PARM Values

The following discusses the PARMs on the EXEC statement for SASSBK00.



Where:

OPT

Indicates which functions are desired. This is a required parameter. Must contain at least one of the following values:

BACKUP

Performs a database copy function. Required if the backup function is desired. Mutually exclusive with RELOAD. BACKUP output is to ddname BACKUP. All database data sets are written to the backup data set.

RELOAD[FORCE]

Performs a database restore function. Input is from ddname BACKUP. Mutually exclusive with BACKUP. RELOAD is meaningless unless BACKUP has been done previously with SASSBK00. FORCE indicates that certain key edit errors are to be allowed and indicated database elements are to be reloaded. Normally, no element with a key edit error is reloaded, but the FORCE option overrides to cause such elements to RELOAD. This parameter is only valid with RELOAD.

LIST

Creates an inventory list of database contents. See report SASSBK07-01, on Field Descriptions - SASSBK07-01 on page 7-24. The output is to ddname DBLIST. LIST may be specified by itself to produce only the list using the live database on DASD. It may be specified in combination with BACKUP or RELOAD functions to produce a list corresponding to the activity specified. LIST is an optional subparameter and mutually exclusive with the SUMM subparameter.

SUMM

Creates only a summary version of report SASSBK07-01. SUMM is mutually exclusive with LIST and may be used with either BACKUP or RELOAD functions as well as being specified by itself. This is the default.

XREF

Creates database cross-reference reports, SASSXREF-01 and SASSXREF-02. XREF may be specified by itself to produce the reports using the live database on DASD. It may be specified in combination with either RELOAD or BACKUP functions to produce reports corresponding to these functions. It may also be specified with DB=TAPEONLY if only reports are to be produced without any other function. The tape used for this must have been created, by a previous BACKUP function, with SASSBK00.

Note: When using XREF by itself, a BACKUP DD is still required, but may point to a temporary data set.

DB

Indicates what organization is used for the CA-7 database. This is a required parameter and the value must be one of the following:

TAPEONLY

Indicates only the backup tape is to be used to generate reports. TAPEONLY is valid with OPT parameter values XREF, SUMM, or LIST.

ALLVSAM

Indicates the database is totally VSAM as defined in the control statements residing in the DBPARMS member of the CA-7 JCLLIB.

EXITMOD

Indicates the name of a user exit module to which all database records are passed as they are read or written from or to the tape data set. When EXITMOD is used with OPT=RELOAD or DB=TAPEONLY, the specified module receives each record read from tape. When EXITMOD is used with OPT=BACKUP, the module specified receives each record to be written to tape. Only valid when OPT values BACKUP or RELOAD or DB=TAPEONLY specified. See 9.4, "Standard Exit Descriptions" on page 9-33 for details on the use of this exit.

XREFMOD

Indicates the load module used to produce cross-reference reports. The default module is SASSXREF which is provided with the product. XREFMOD is only required when OPT=XREF is specified and some module name other than SASSXREF produces the cross-reference reports.

7.1.2 Reload Procedures

Considering recovery needs, the overriding concern is to restore the database quickly and cleanly. Remember that without a database, CA-7 (and scheduling) are stopped. Therefore, the fastest utility is the one to plan for and you must have used it for your backups before you can recover. For the recovery to be clean, all database files must be reloaded from the same backup run; otherwise, internal references between these files are in error. Never recover only one or two of the database files.

The following descriptions present some common utilities, but by no means, all available utilities.

7.1.2.1 IDCAMS

The following illustrates recovering an ALLVSAM database which was backed up using IDCAMS. If the VSAM cluster was lost, you have to DEFINE the files before doing the REPRO. You may wish to DELETE and DEFINE the files anyway to obtain space reorganization and shorten path lengths for index records. If space allocation or target packs for the files are to change, the DELETE and DEFINE are required. Also, any changes that require a DEFINE should be checked against, and similar changes made to, the DBPARMS definitions used by CA-7.

```
//jobname JOB .....
//*
//IDCAMS EXEC PGM=IDCAMS
//SYSIN DD *
  REPRO INFILE(FILE1) OUTDATASET(your.IDS.name) REPLACE
  REPRO INFILE(FILE2) OUTDATASET(your.SASDS.name) REPLACE
  REPRO INFILE(FILE3) OUTDATASET(your.SASJOB.name) REPLACE
//SYSPRINT DD SYSOUT=*
//FILE1 DD DISP=OLD,
//      DSN=backup.IDS.name, <== You may want to use a GDG
//      UNIT=tape, <== Specify desired unit type
//      VOL=(,RETAIN),
//      LABEL=1
//FILE2 DD DISP=OLD,
//      DSN=backup.SASDS.name, <== You may want to use a GDG
//      UNIT=AFF=FILE1,
//      VOL=(,RETAIN,REF=*.FILE1),
//      LABEL=2
//FILE3 DD DISP=OLD,
//      DSN=backup.SASJOB.name, <== You may want to use a GDG
//      UNIT=AFF=FILE1,
//      VOL=REF=*.FILE1,
//      LABEL=3
```

Note: This example should be checked thoroughly to ensure that it works in your specific environment.

7.1.2.2 CA-ASM2

Sample JCL to recover using backup files created by CA-ASM2 is shown below. See the CA-ASM2 documentation for more information and descriptions of commands and options. The JCL procedure indicated probably has a different name at your installation.

```
//jobname JOB .....  
//*  
//RECOVER EXEC ASM2CMDU  
//SYSIN DD *  
$RB DSNNAME('your.IDS.name')  
$RB DSNNAME('your.SASDS.name')  
$RB DSNNAME('your.SASJOB.name')  
$RX TYPE(BKUP)  
/*
```

Note: This example should be checked thoroughly to ensure that it works in your specific environment.

7.1.2.3 DFDSS

The following is sample JCL for a recovery using DFDSS. As with IDCAMS, if there is a problem with the VSAM cluster, you have to DELETE and DEFINE the files before the recovery. Also, if the target packs are to change, you must use IDCAMS to DELETE and DEFINE the files. Likewise, to obtain reorganized file space, you may wish to DELETE and DEFINE anyway.

```
//jobname JOB .....
//*
//RECOVER EXEC PGM=ADRDSSU
//TAPE DD DISP=OLD,
//      DSN=your.backup.name <== Could be GDG
//DASD DD DISP=SHR,
//      UNIT=disk, <== Specify your unit type
//      VOL=SER=xxxxxx <== Volume containing database
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
        RESTORE INDDNAME(TAPE) OUTDDNAME(DASD) REPLACE -
        DATASET(INCLUDE(your.IDS.name, -
                        your.SASDS.name, -
                        your.SASJOB.name))
/*
```

Note: This example should be checked thoroughly to ensure that it works in your specific environment.

7.1.2.4 SASSBK00

The SASSBK00 program can be used not only to back up the database, but also to reload it. However, a backup performed by SASSBK00 must precede a reload.

7.1.2.5 Database Reload JCL Considerations

The JCL for a database reload is very similar to a database backup. See the previous discussion and 7.1.1.4, “SASSBK00” on page 7-6.

The differences between typical JCL for backup and reload are listed below and apply to the reload only:

- OPT=RELOAD must be specified on the PARM list. RELOAD performs a database tape-to-disk function. Input is from ddname BACKUP. The RELOAD subparameter is mutually exclusive with the BACKUP subparameter and is meaningless unless a BACKUP has been done previously with SASSBK00.
- The ddname BACKUP is used by the program for the reload function. The DISP for the data set must be OLD when RELOAD is specified as a subparameter.
- The DISPs for ddnames UCC7JLIB, UCC7DLIB, UCC7IDS must be OLD when RELOAD is specified.
- If the database is on SMS controlled volumes, then it may be necessary to add an extra IDCAMS step in front of the reload step to delete the 3 database files. The problem occurs when there are multiple volumes within a SMS storage class.
- If a RELOAD is being done for a different volume, it is necessary to add an extra IDCAMS step in front of the reload step to delete the three database files.

7.1.3 DBPARMS Parameter Deck

The DBPARMS parameter deck is a card-image deck used to define the VSAM data sets. This deck must be provided through the ddname of DBPARMS.

Once the data sets are loaded, the same DBPARMS deck must be coded in the JCL for CA-7 with any execution of the Batch Card Load program (SASSBCLP), and all uses of the Database Verification program (UCC7DBVR). For new users of CA-7, the SASSCDSI program used to initialize the database also requires this data set. To satisfy this requirement for multiple references to the same DBPARMS information, this deck should be made a member of a common card-image PDS. Shared access from all modules can then be easily provided while ensuring that all also use the same database specifications.

UCC7DBASE statements in the DBPARMS deck contain the specifications for the VSAM data sets and some performance options to be used.

A sample DBPARMS deck is created during the SYSGEN process of CA-7. This is probably the easiest way to generate the DBPARMS data set. It can then be changed, if necessary. This deck resides in member DBPARMS in the CA-7 JCLLIB created during the CA-7 Sysgen process.

7.1.4 UCC7DBASE Statements

These statements are included in the DBPARMS deck to define the VSAM data sets in an ALLVSAM database.

One UCC7DBASE statement is included for **each** VSAM data set. These statements are identified by the keyword UCC7DBASE which must appear in position 2 of the record.

Following the UCC7DBASE keyword, and at least one blank, the definition of a data set is coded. These specifications are in IDCAMS command language format. Continuations are necessary due to the number of parameters required.

7.1.4.1 Syntax

```
UCC7DBASE      (DATATYPE(xxxx,xxx,...,xxx) -
                STRINGS(nn,nn,...,nn) -
                BUFNI (nn) -
                BUFND (nn) -
                ALLOCJCL | ALLOCDYN -
                DDNAME (ddddddd) -
                IDCAMSDEFINE (DEFINE CLUSTER...))
```

Where:

UCC7DBASE

Identifies the statement as defining one VSAM data set in the logical CA-7 database. Should begin in position 2 of the record and be followed by at least one blank. One UCC7DBASE statement is required for each VSAM data set.

DATATYPE

Identifies the data type(s) which are in this data set. Each data type represents a unique record format (or type) in the database. Each data type code must be specified once, but only once, as a DATATYPE subparameter. All data types must be coded whether or not they are actually used. Trailing blanks in data type codes may be omitted here. The CA-7 reload function places data in the data sets based on the DATATYPE values. See 7.1.4.2, “UCC7DBASE Data Types” on page 7-18 for a definition of the data type code values and what they represent.

STRINGS

Defines the number of access strings (VSAM RPLs) to be provided for each data type defined with the DATATYPE parameter. String values within parentheses are positional and correspond to the sequence of the data types defined in the DATATYPE list. One and only one value must be provided for each data type. Value is numeric and may be from 1 to 99. Values may be varied as desired to attain the desired performance level, without having to reload the database. A value of 1 (the default) is adequate for backups, reloads and other batch utilities, but a value of 3 or greater should be used for CA-7 to avoid lockout conditions.

BUFNI

Defines the number of buffers to be used for processing VSAM index entries for the data set. May be from 1 to 99. Default is equal to number of VSAM index levels. Values may be varied as desired to attain the desired performance level, without having to reload the database.

Note: To improve performance, it is recommended that BUFNI be at least the value of the STRINGS value + 5.

BUFND

Defines the number of buffers to be used for processing VSAM data entries in this data set. May be from 1 to 99. Default is equal to the STRINGS values for the data types in this data set. Values may be varied as desired to attain the desired performance level, without having to reload the database.

Note: To improve performance, it is recommended that BUFNO be at least the value of the STRINGS value + 5.

ALLOCJCL | ALLOCDYN

Indicates whether allocation of the data set is to be performed through JCL or dynamically. Only one may be specified. If ALLOCJCL is specified, DDNAME must also be specified. Default is ALLOCDYN.

DDNAME

Indicates the ddname to be used for the data set. Should be UCC7JLIB, UCC7DLIB or UCC7IDS. See 7.1.4.2, “UCC7DBASE Data Types” on page 7-18 to determine which one to use. DDNAME should only be specified when ALLOCJCL is used.

IDCAMSDEFINE

Provides the IDCAMS parameters necessary to define the VSAM cluster which contains the types of data specified in the DATATYPE parameter. Parameters coded within the parentheses are passed to IDCAMS, exactly as coded, by the CA-7 reload modules to create the data sets. See 7.1.4.3, “IDCAMSDEFINE Parameters” on page 7-19 for considerations unique to CA-7. Also see the DBPARMS deck generated by Stage I installation. These values must not change except during a reload of the database.

7.1.4.2 UCC7DBASE Data Types

UCC7DBASE defining	DDNAME	DATATYPE values*	Equivalent PDS database records
Job data set	UCC7JLIB	JBD	Job data in directory records
		JBM	Job data in member records
Dataset data set	UCC7DLIB	DSD	DSN data in directory records
		DSM	DSN data in member records
		NWD	Network data in directory records
		NWM	Network data in member records
		PPD	Documentation data in directory records
		PPM	Documentation data in member records
		SID	Input workstation network schedule data in directory records
		SIM	Input workstation network schedule data in member records
		SJD	Job schedule data in directory records
		SJM	Job schedule data in member records
		SOD	Output workstation network schedule data in directory records
		SOM	Output workstation network schedule data in member records
Index data set	UCC7IDS	I	All index data set entries
Note: * A data type of CNTL also resides in each data set. The user should <u>not</u> code a DATATYPE subparameter of CNTL in the UCC7DBASE statements. These are used for internal control information and are managed exclusively by the CA-7 VSAM subsystem.			

7.1.4.3 IDCAMSDEFINE Parameters

The IDCAMSDEFINE parameters, beginning with the keyword DEFINE and through the end of the IDCAMSDEFINE parameters, are coded according to IBM IDCAMS conventions to cause IDCAMS to create the data sets in the desired way. These values are passed to IDCAMS during the load/reload process. Options taken are then used, for all accesses to the data sets, by CA-7 and its related batch-only programs.

Values for some of the IDCAMS parameters have considerations unique to CA-7.

INDEXED must be specified.

KEYS must specify (52 0).

IMBED is recommended (unless cache is used).

CONTROLINTERVALSIZE should specify 4096 for the DATA component and 4096 for the INDEX component.

RECORDSIZE parameter determines the record lengths in the data sets. The maximum record size is selected by the user but cannot exceed 32760 or be smaller than 3122. Recommended average and maximum values for the data sets are as follows:

- Job data set - average 60, maximum 4080
- Dataset data set - average 60, maximum 4080
- Index data set - average 60, maximum 4080

If necessary, to relieve main memory requirements or to improve performance, these values can be changed. However, if maximum record size is greater than the CI size, the SPANNED attribute must be specified. The SPANNED attribute requires increased memory and CPU overhead and is therefore not recommended.

SHAREOPTIONS should specify (2,3). This allows batch facilities such as BCLP, data-base verification, and so forth, to be performed concurrent with CA-7 execution.

Secondary space should be allocated for all three files.

7.1.5 Backup/Reload Reports

CA-7 DB Backup/Reload Control Report on page 7-20 shows the control report which is produced for each execution of SASSBK00. Field Descriptions - SASSBK07-01 on page 7-24 shows the output from SASSBK00 when the LIST or SUMM OPT parameter is specified.

Cross-reference reports can also be generated from SASSBK00 when OPT=XREF is specified.

SASSBK00-01	C A - 7 D B B A C K U P / R E L O A D	PAGE 1
A DATE: 10-13-yy/yy.287	B TIME: 10:54:37	CONTROL REPORT
FUNCTIONS REQUESTED: OPT=BACKUP,DB=ALLVSAM		C
FUNCTIONS PERFORMED: BACKUP: YES RELOAD: NO		D LIST: NO XREF: NO
USER EXIT USED: USER EXIT: N/A		VOLSER NO.- N/A
LIBRARY NAME: E N/A		H
F DBPARMS DATASET USED:		
I		
J DDNAME: DBPARMS	K DATASET NAME: SSDDEV.CA07.JCLDS1(DBPARMS)	L VOLSER NO.- M80008 M DSORG=PDS
N	UCC7DBASE (DATATYPE(I) - STRINGS(3) BUFNI(5) BUFND(5) - ALLOCDYN - IDCAMSDEFINE (- DEFINE CLUSTER (- NAME (V.SSDDEV.CA07.IDS) - IXD IMBED NREPL SPEED - KEYS(52 0) - RECORDSIZE(60 4080) - VOL(M80008) TRK(3 1) - SHR(2 3)) - DATA (NAME(V.SSDDEV.CA07.IDS.DATA) - CONTROLINTERVALSIZE(4096) - UNIQUE) - INDEX (NAME(V.SSDDEV.CA07.IDS.INDEX) - CONTROLINTERVALSIZE(4096) - UNIQUE))) UCC7DBASE (DATATYPE(- DSD,DSM,NWD,NWM,SJD,SJM,SID,SIM,SOD,SOM,PPD,PPM) - STRINGS(3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3) BUFNI(38) BUFND(38) - ALLOCDYN - IDCAMSDEFINE (- DEFINE CLUSTER (- NAME (V.SSDDEV.CA07.SASDS) - IXD IMBED NREPL SPEED - KEYS(52 0) - RECORDSIZE(60 4080) - VOL(M80008) TRK(5 1) - SHR(2 3)) -	

Figure 7-2. SASSBK00-01 CA-7 DB Backup/Reload Control Report (Page 1)

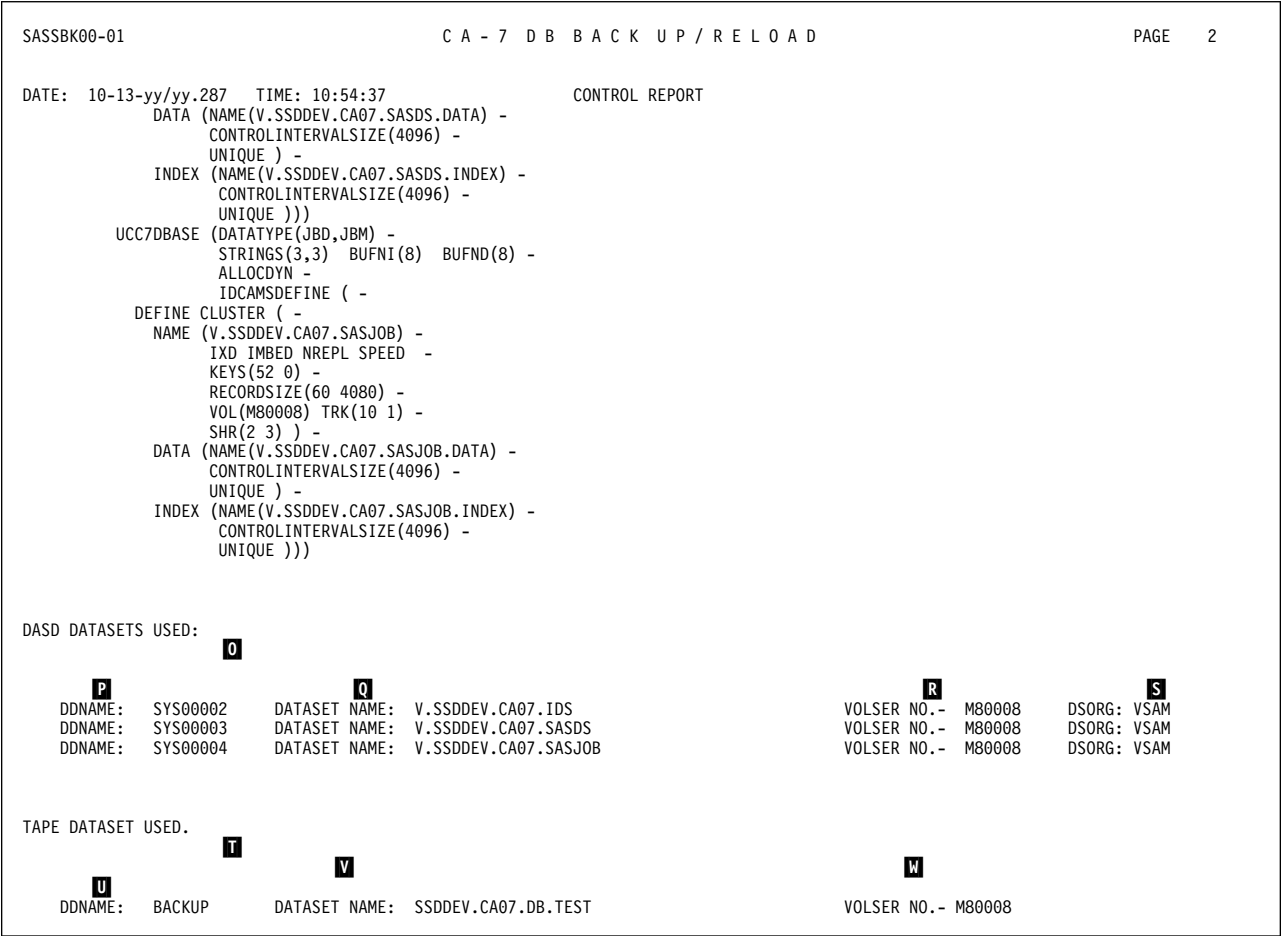


Figure 7-3. SASSBK00-01 CA-7 DB Backup/Reload Control Report (Page 2)

This report indicates what was requested, what was performed, and the data sets used to accomplish whatever functions were requested.

7.1.5.1 Fields

- A** The date of the run shown in both Gregorian and Julian formats: mm-dd-yy and yyddd.
- B** The time-of-day of the run.
- C** Shows the PARM data from the EXEC statement.
- D** Shows which functions were performed by the program.
- E** If the user specified an exit, the exit name is listed on the line below.
- F** The user exit module name.
- G** If an exit was specified, this is the name of the library in which the exit program resides.

- H** If the user specified an exit, this is the volume serial number on which the exit program/library resides.
- I** The DBPARMS deck used.
- J** The ddname used to access DBPARMS.
- K** The deck name in which DBPARMS resided.
- L** Volume serial number on which the DBPARMS deck resides.
- M** Data set organization of the DBPARMS deck.
- N** The UCC7DBASE control statements which define the database.
- O** The database data sets used.
- P** The ddnames used for the database data sets.
- Q** The data set names of the database data sets.
- R** Volume serial number on which the database data sets reside.
- S** Data set organization of each database data set.
- T** The tape data set used.
- U** The ddname used for any tape data set (BACKUP).
- V** The data set name of any tape data set used.
- W** Volume serial number(s) of the tape data set used.

SASSBK07-01		C A - 7 D B B A C K U P / R E L O A D		PAGE 25	
DATE: 10-12-yy/yy.286		TIME: 11.21.38		DATA BASE INVENTORY	
--ELEMENT IDENTITY -----		# RECORDS		DATA TYPE: SID I/P SCHED DIRECTORY	
TOTAL SID ELEMENTS: 2		TOTAL # RECORDS: 2			
--ELEMENT IDENTITY -----		# RECORDS		DATA TYPE: SIM I/P SCHED MEMBER	
SI000000		1			
SI000005		1			
TOTAL SIM ELEMENTS: 2		TOTAL # RECORDS: 2			
--ELEMENT IDENTITY -----		# RECORDS		DATA TYPE: SJD JOB SCHED DIRECTORY	
SJ000000		1			
SJ000001		1			
SJ000004		1			
SJ000005		1			
SJ000022		1			
TOTAL SJD ELEMENTS. 5		TOTAL # RECORDS. 5			
--ELEMENT IDENTITY -----		# RECORDS		DATA TYPE: SJM JOB SCHED MEMBER	
SJ000000		1			
SJ000001		1			
SJ000004		1			
SJ000005		1			
SJ000022		1			
TOTAL SJM ELEMENTS. 5		TOTAL # RECORDS. 5			
--ELEMENT IDENTITY -----		# RECORDS		DATA TYPE: SOD O/P SCHED DIRECTORY	
S0000000		1			
S0000002		1			
TOTAL SOD ELEMENTS. 2		TOTAL # RECORDS. 2			
--ELEMENT IDENTITY -----		# RECORDS		DATA TYPE: SOM O/P SCHED MEMBER	
S0000000		1			
S0000002		1			
TOTAL SOM ELEMENTS. 2		TOTAL # RECORDS. 2			
DATA BASE TOTALS. 1,291		TOTAL # RECORDS. 4,203			

Figure 7-4. SASSBK07-01 CA-7 Database Inventory Report

This report lists information on what data sets constitute the database, the data types each data set contains, record counts and optionally (with OPT=LIST) lists the element identities (VSAM keys) of each element in the database.

7.1.5.2 Fields

- A** The date of the run shown in both Gregorian and Julian formats: mm-dd-yy and yyddd.
- B** The time-of-day of the run.
- C** The individual element names in the data set being inventoried. (Not listed if SUMM was specified.)
- D** Number of records for each element.
- E** The data type code for the elements in the specified data set.
- F** The total number of elements in this data type.
- G** The total number of records for this data type.
- H** The total number of database elements in the entire database.
- I** The grand total of records in the entire database.

7.2 ARF Backup and Reload

Because the ARF file is a VSAM KSDS there are several utilities that can be used for backup and reload. Two procedures using IDCAMS are provided as part of the installation process and reside on the CA-7 JCL library. However, use of these procedures is not required. Alternatives to IDCAMS such as CA-ASM2 and DFDSS may be used if desired.

7.2.1 ARF Backup Procedure - CA7ARK

The ARF backup procedure CA7ARK is an execution of IDCAMS to back up the contents of the ARF file to tape:

```

/*-----***
/*  CA-7 ARF DATA BASE BACKUP JCL PROCEDURE      ***
/*-----***
//CA7ARK PROC OUT='*',
//          ARFIN='CAI.CA7.ARF',
//          BUNIT='TAPE9',
//          BLABEL='LABEL=(1,SL)',
//          BDSN='CAI.CA7.ARF.BACKUP'
/*-----***
/*  IDCAMS REPRO ARF DATA BASE TO SEQUENTIAL FILE ***
/*-----***
//AREPRO EXEC PGM=IDCAMS,REGION=512K
//INPUT  DD DSN=&ARFIN,
//          DISP=SHR
//OUTPUT DD DSN=&BDSN,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=&BUNIT,&BLABEL
//          DCB=(RECFM=VB,LRECL=32700,BLKSIZE=32704)
//SYSUDUMP DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT,DCB=BLKSIZE=133
//SYSIN   DD DISP=SHR,
//          DSN=CAI.CA7.JCLLIB(REPRO)

```

7.2.1.1 CA7ARK JCL Procedure DD Statements

AREPRO Step:

INPUT Specifies the CA-7 ARF database component.

OUTPUT Identifies the sequential backup file.

SYSPRINT

Used for message output.

SYSIN Refers to a member on the CA-7 installation JCL library containing IDCAMS control statements.

7.2.2 ARF Reload Procedure - CA7ARL

The ARF reload procedure CA7ARL is an execution of IDCAMS to reload that will cause the ARF file to be deleted, redefined, and loaded from a tape created by the backup procedure:

```

/*-----***
/* CA-7 ARF DATA BASE DELETE/DEFINE/RELOAD PROC ***
/*-----***
//CA7ARL PROC OUT='*',
//      ARFOUT='CAI.CA7.ARF',
//      BUNIT='TAPE',
//      BLABEL='LABEL=(1,SL)',
//      BDSN='CAI.CA7.ARF.BACKUP'
//*
//ARFLOAD EXEC PGM=IDCAMS,REGION=512K
//INPUT  DD DSN=&BDSN,
//        DISP=SHR,
//        UNIT=&BUNIT,&BLABEL
//        DCB=(BLKSIZE=32704)
//OUTPUT DD DSN=&ARFOUT,
//        DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN  DD DISP=SHR,
//        DSN=CAI.CA7.JCLLIB(ARFDEL)
//        DD DISP=SHR,
//        DSN=CAI.CA7.JCLLIB(ARFALLOC)
//        DD DISP=SHR,
//        DSN=CAI.CA7.JCLLIB(REPRO)

```

7.2.2.1 CA7ARL JCL Procedure DD Statements

ARFLOAD Step:

INPUT Specifies the sequential file created by the backup job.

OUTPUT Specifies the CA-7 ARF file.

SYSPRINT

Used for message output.

SYSIN Points to the following members on the CA-7 installation JCL library:

ARFDEL Contains IDCAMS control statements to delete the ARF file.

ARFALLOC

Contains IDCAMS control statements to allocate the ARF file.

REPRO Contains IDCAMS control statements to reload the ARF file with data from the backup file.

7.2.3 ARF Reload Special Considerations

Each production job that is being monitored by ARF is said to be under the control of an ARFSET. An ARFSET is a collection of ARF "rules" which define the recovery conditions for a job or a set of jobs. When a job using an ARFSET enters the request queue, the name of the ARFSET is recorded in the queue. An "active" copy of the ARFSET will also be recorded on the ARF file. When this active ARFSET copy is created, a logical pointer to the copy is also recorded in the queue.

When CA-7 is restarted, ARFSET pointers in the trailer queue are read to relocate the active ARFSET copies that are used for ARF monitoring. If these cannot be found, then the jobs will be left in manual recovery status (as if no ARFSET name had been supplied).

The IDCAMS parameters used to define the ARF file are described in the following topic.

7.2.3.1 ARF Data Set IDCAMS Define Parameters

IDCAMS define parameters for the ARF data set component:

Recordsize(400 32700)

This parameter determines the record length in the data set. These values should not be changed.

SHR(2 3)

Share options specify concurrent read access of the data set is allowed. Recommended.

IMBED

This parameter relates to sequence-set record placement. Recommended.

REPLICATE

This parameter relates to each index record written on a track. Recommended.

FREESPACE

Specifies the amount of space to be left free when the cluster is loaded (by percentages). Recommend 20, 10.

UNIQUE

Specifies unique space for the data and index portions of the data set. Recommended.

KEYS(53, 0) Required values are 53, 0.

7.3 VRM Backup and Reload

The Virtual Resource Management (VRM) facility provides backup and reload procedures to ensure data recovery in the event of a system failure. The VRM backup/reload JCL procedures CA7VBK and CA7VRL are generated during the installation process and reside in the CA-7 JCL library.

7.3.1 VRM Backup Procedure

The VRM backup procedure CA7VBK consists of the following steps:

Steps:

VREPRO Reads the VRM database component (ddname = INPUT) and creates a sequential file (ddname = OUTPUT) using the IBM utility IDCAMS.

VBUILD Reads the sequential file created in Step 1 and rebuilds the VRM resource type R cross-reference records and generates a report of all records found on the VRM database component. See the sample report.

7.3.1.1 VRM Backup Procedure - CA7VBK

```

/*-----*
/*          CA-7 VRM DATA BASE BACKUP JCL PROCEDURE - CA7VBK          *
/*-----*
/*CA7VBK PROC OUT='*',
/*      UNIT=3380,
/*      LOADLIB='CAI.CA7.CAILIB',
/*      VRMIN='CAI.CA7.VRM',
/*      BUNIT='TAPE',
/*      BLABEL='LABEL=(1,SL)',
/*      BDSN='CAI.CA7.VRM.BACKUP'
/*-----*
/*          IDCAMS REPRO VRM DATA BASE COMPONENT TO SEQUENTIAL FILE    *
/*-----*
//VREPRO EXEC PGM=IDCAMS,REGION=512K
//INPUT  DD DSN=VRMIN,
//        DISP=SHR
//OUTPUT DD DSN=VRMWORK,
//        DISP=(NEW,PASS),
//        UNIT=UNIT,
//        SPACE=(CYL,(5,5)),
//        DCB=(RECFM=VB,LRECL=32700,BLKSIZE=32704)
//SYSPRINT DD SYSOUT=OUT
//SYSIN   DD DISP=SHR,
//        DSN=CAI.CA7.JCLLIB(REPRO)
/*-----*
/*          REBUILD OF VRM RESOURCE RECORDS                             *
/*-----*
//VBUILD EXEC PGM=SASSVDB0,REGION=1M
//STEPLIB DD DSN=LOADLIB,
//        DISP=SHR
//VRMIN   DD DSN=VRMWORK,
//        DISP=(OLD,DELETE,DELETE)
//VRMOUT  DD DSN=BDSN,
//        DISP=(NEW,CATLG,DELETE),
//        UNIT=BUNIT,BLABEL
//        DCB=(RECFM=VB,LRECL=32700,BLKSIZE=32704)
//SYSUDUMP DD SYSOUT=OUT
//SYSPRINT DD SYSOUT=OUT,DCB=BLKSIZE=133
//SYSOUT  DD SYSOUT=OUT
//SORTCNTL DD DISP=SHR,DSN=CAI.CA7.JCLLIB(VRMSORT)
//SORTWK01 DD UNIT=UNIT,SPACE=(CYL,5)
//SORTWK02 DD UNIT=UNIT,SPACE=(CYL,5)
//SORTWK03 DD UNIT=UNIT,SPACE=(CYL,5)
//SORTWK04 DD UNIT=UNIT,SPACE=(CYL,5)
//SORTWK05 DD UNIT=UNIT,SPACE=(CYL,5)
//SORTWK06 DD UNIT=UNIT,SPACE=(CYL,5)

```

Since VRM records have a maximum length of 32,700, the LRECL and BLKSIZE parameters in the example should not be reduced.

7.3.1.2 CA7VBK JCL Procedure DD Statements

VREPRO Step:

INPUT Specifies the CA-7 VRM database component.

OUTPUT Identifies the sequential file created for input into the VBUILD step.

SYSPRINT

Used for message output.

SYSIN References the IDCAMS control statement member on the CA-7 installation JCL library.

VBUILD Step:

STEPLIB Must specify the CA-7 load library.

VRMIN Specifies the sequential file created in the VREPRO step.

VRMOUT

Specifies a sequential backup file which contains the VRM database component records to be used during the VRM reload process.

7.3.1.3 SASSVDB0 Sample Report

SASSVDB0 Record Totals for CA-7 VRM database on yy.ddd at hh:mm:ss.tt

Total number of F (X'C6')	- CA-7 Active CPM Records	:	0
Total number of J (X'D1')	- CA-7 VRM Job	:	5
Total number of P (X'D7')	- CA-7 User Profile	:	2
Total number of R (X'D9')	- CA-7 VRM Dataset XREF (Rebuilt)	:	10
Total number of U (X'E4')	- CA-7 VRM Manually Freed Resource Count Resource	:	0
Total number of V (X'E5')	- CA-7 VRM Manually Freed Resource	:	0
Total number of W (X'E6')	- CA-7 VRM Corequisite Resource	:	1
Total number of X (X'E7')	- CA-7 VRM Active Job	:	0
Total number of Y (X'E8')	- CA-7 VRM Active Resource Count Resource	:	0
Total number of 9 (X'F9')	- CA-7 VRM Database Initialization Record	:	1
Total Number of Records on VRMOUT		:	19
Total Number of CA-7 VRM Dataset XREF Records bypassed on VRMIN		:	10

7.3.2 VRM Reload Procedures

You can reload the VRM backup data set, created using the VRM backup procedure CA7VBK, using the VRM reload procedure CA7VRL. This procedure uses the IBM IDCAMS utility to delete, reallocate, and reload the VRM database component. See 7.3.3, “VRM Reload Special Considerations” on page 7-32 for additional information.

7.3.2.1 VRM Reload Procedure JCL

```

/*-----*
/*      CA-7 VRM DATA BASE DELETE/DEFINE/RELOAD PROC - CA7VRL      *
/*-----*
//CA7VRL PROC OUT='*',
//      VRMOUT='CAI.CA7.VRM',
//      BUNIT='TAPE',
//      BLABEL='LABEL=(1,SL)',
//      BDSN='CAI.CA7.VRM.BACKUP'
/*-----*
/*      RELOAD THE VRM DATA BASE COMPONENT      *
/*-----*
//VRMLoad EXEC PGM=IDCAMS,REGION=512K
//INPUT    DD DSN=&BDSN,
//          DISP=SHR,
//          UNIT=&BUNIT,&BLABEL
//OUTPUT    DD DSN=&VRMOUT,
//          DISP=SHR
//SYSPRINT  DD SYSOUT=&OUT
//SYSIN     DD DISP=SHR,
//          DSN=CAI.CA7.JCLLIB(VRMDEL)
//          DD DISP=SHR,
//          DSN=CAI.CA7.JCLLIB(VRMALLOC)
//          DD DISP=SHR,
//          DSN=CAI.CA7.JCLLIB(REPRO)
//*

```

7.3.2.2 CA7VRL JCL Procedure DD Statements

INPUT Specifies the sequential file created in the CA7VBK procedure VBUILD step.

OUTPUT Specifies the CA-7 VRM VSAM database component.

SYSPRINT

Used for message output.

SYSIN

Points to the following CA-7 JCL library members:

VRMDEL IDCAMS control statements to delete the VRM database component.

VRMALLOC

IDCAMS control statements to allocate the VRM database component.

REPRO

IDCAMS control statements to repro the CA7VBK backup data set into the VRM database component.

7.3.3 VRM Reload Special Considerations

The VRM database component contains both static and active records. The static records maintain information about job to resource connections. The active records are used to track resource use by jobs during the normal job submission and job completion process.

The VRM backup process backs up ALL records on the database to the backup data set. If it is necessary to reload the VRM database component from an old backup data set, a CA-7 cold type of start (COLD, FORM, MOVQ) must be performed to delete any active resource records reloaded from the old backup data set. This ensures synchronization between active job resource use and the queue records.

If CA-7 is shut down and a VRM backup/reload is done to reorganize the VRM file, then CA-7 can be restarted with a WARM or ERST start. This is because the active VRM records are still synchronized with the queues.

7.3.3.1 VRM Data Set IDCAMS Define Parameters

IDCAMS define parameters for the VRM data set component:

Recordsize(400 32700)

This parameter determines the record length in the data set. These values should not be changed.

SHR(2 3)

Share options specify concurrent use of the data set is allowed. Recommended.

IMBED

This parameter relates to the sequence-set record placement. Recommended.

REPLICATE

This parameter relates to each index record written on a track. Recommended.

FREESPACE

Specifies the amount of space to be left free when the cluster is loaded (by percentages). Recommend 20,10.

UNIQUE

Specifies unique space for the data and index portions of the data set. Recommended.

KEYS(53, 0) Must be as specified 53, 0. Required.

7.4 Terminals

The following paragraphs contain suggestions for recovering from unresponsive or "hung" terminals connected to CA-7.

The following steps and their associated commands are only for terminals that have encountered an error and are subsequently "hung." A terminal with a CA-7 command in process cannot be interrupted. To do so could result in CA-7 abending.

7.4.1 VTAM Terminals

The following steps are suggested procedures for a hung terminal connected to CA-7.

1. Find another terminal defined to CA-7 and log on.
2. Enter /STOP,T=termname

Where termname is the name of the hung terminal as it is defined in the initialization file, the TERM statement, NAME parameter.

3. Enter /START,T=termname

Where termname is the same terminal name used in Step 2.

If the terminal is still hung, try the following:

1. Enter /STOP,T=termname

Where termname is the name of the hung terminal as it appears in the initialization file, the TERM statement, NAME parameter.

2. Enter /CLOSE,T=termname

Where termname is the same terminal name used in Step 1. /CLOSE disconnects the terminal from CA-7. If the terminal name is not specified, the terminal issuing the command is disconnected from CA-7. The operator has to reconnect and log on again.

3. Enter /PURGPG,LT=termname

Where termname is the same terminal name used in Step 1.

4. Enter /OPEN,T=termname

Where termname is the same terminal name used in Step 1.

5. Enter /START,T=termname

Where termname is the same terminal named used in Step 1.

Note: The commands listed in the steps can be issued through a batch terminal.

After Step 3 (/PURGPG), the terminal may need to be varied active through VTAM.

If the suggested steps have been tried and the terminal is still hung, then shut down and reinitialize CA-7.

If VTAM is down and CA-7 is up, CA-7 can be shut down and restarted after VTAM has been restored. Or after VTAM has been reinstated, the following suggested steps may be used. The commands in these steps can be issued from a batch terminal or from the system console if the console is defined and opened to CA-7; that is, CA-7 has an outstanding WTOR.

1. Enter /CLOSE,GROUP=groupname

Where groupname is the name of the first group of VTAM terminals defined to CA-7 in the initialization file. This command disconnects all VTAM terminals from CA-7.

2. Enter /OPEN,GROUP=groupname

Where groupname is the same as that specified in Step 1. This command reconnects CA-7 to VTAM.

7.4.2 Batch Terminals

The only way to recover a batch terminal is to shut down CA-7, cancel any batch terminal interface programs, and restart CA-7. Any attempt to recover a batch terminal currently in use could result in an abend of CA-7.

Batch terminal interface jobs should never be canceled while CA-7 is active. Canceling a BTI job does not prevent CA-7 from processing the commands.

7.5 Workload Considerations

The following are some of the general considerations for ensuring that the production workload suffers only minimal disruptions for most disaster situations.

To preserve the integrity and effectiveness of CA-7 in the most severe disaster recovery situations, good practices regarding backup of the system are imperative.

It is not just CA-7 backup and recovery procedures at question, but also all the application systems that CA-7 controls and all the systems that are needed to maintain an orderly work flow.

7.5.1 CA-7 Considerations

Recovery procedures must include jobs to back up and reload the systems that are under control of CA-7. These jobs should be under control of CA-7 so they are backed up with CA-7. The user can develop a series of jobs so that one initial backup/reload job can be DEMANDED at the appropriate time, and it triggers the other backup/reload jobs in a logical sequence.

The safest procedure to adopt is frequent backups of most CA-7 data sets and all systems under the control of CA-7. This includes all JCL, load module and control statement data sets, and so forth. All should be backed up on the same time period. During the backup of the CA-7 database, CA-7 should be shut down for totally predictable results.

If frequent backups of CA-7 data sets are not feasible, then the user may wish to consider relating backup frequency to update frequency. For example, a database usually has the highest frequency of update, so it would have the highest frequency of backup. Some data sets have a very low frequency of update in some data centers.

7.5.2 Other CA Products

User backup procedures for CA-7 should take into consideration the disaster recovery aspect. If the user also has CA-11 or CA-1, then CA-7 either interfaces with these or controls them. Therefore, their backups should be timely and synchronized with that of the CA-7 backup.

7.5.3 Other Vendor Software Products

Backups for any other product or software package that are necessary for an orderly progression of work should be under the control of CA-7 so that the backups, restores, and recovery activities are facilitated. Backups of these data sets should follow vendor recommendations.

7.5.4 CA-7 Workload Documentation Considerations

One of the important features of CA-7 that is sometimes overlooked by the users is the workload documentation function. If the workload documentation function is used effectively, the documentation for all relevant systems and recovery requirements for any production consideration may be kept online. If a disaster situation arises, the information is there to aid in accomplishing a recovery. Since the information resides in the database, any backup of the database also preserves this data.

7.5.5 CA-7 NCF Considerations

If the user has CA-7 NCF installed and a disaster situation has occurred, the user could go to a new site with less computing capacity than the destroyed site. The reason is that NCF can offload, to yet another site, the work that cannot be handled in the temporary data center. Some JCL changes would be involved to handle the routing to the other data center(s).

7.5.6 Magnetic Tape Considerations

Recovery procedures should also consider the FTAPE command which can help identify which tapes would be necessary to reestablish orderly work flow. Database cross-reference reports may also be of great help since they itemize the data set names needed for the jobs.

7.5.7 CA-7 Recovery Aid Considerations

The recovery aid reports of CA-7 are available, through the history reporting facility, in the event one or more queues have been destroyed. The remainder of the system must be intact. This facility can determine the status of the workload that was in the system at the time of failure. It then provides assistance in recovering from such a failure.

Note: In the event of a DASD hardware failure and the VRM database component is lost, a CA-7 COLD start must be performed to resynchronize the VRM resource use records and the CA-7 status queues.

7.6 Recovery Aid

7.6.1 Recovery Aid Facility

Recovery from system failures is normally a fairly easy process with the CA-7 system. A warm start or emergency restart is performed and an LQ display is done to determine the current status of jobs which were active in the CA-7 system at the time of failure. Then maintenance and restarts are performed where necessary.

However, if the failure causes even one CA-7 queue to be destroyed, then it is impossible to warm start or emergency restart CA-7. Since most queue data is lost with a cold start, it becomes very difficult to determine the status of jobs which were active when the system failed, or even to determine which jobs were active when the failure occurred.

The recovery aid facility provides formatted information from log data reflecting major milestones in the life of each CPU job. This information can be helpful in recovering from a CA-7 failure in which one or more of the CA-7 queues are destroyed. Although this feature does not provide automatic queue recovery, it does provide information on all CPU jobs active in CA-7 at the time of failure.

No information is given concerning the state of a workstation network's progress in the preprocessing or postprocessing queue. However, for jobs in the request, ready, active, and prior-run queues, network requirements are noted on a yes or no basis.

7.6.1.1 Queue Milestones

The recovery aid facility uses the batch history reporting facility to produce the recovery aid reports from the following records in the CA-7 log data set:

Type X'64' - CA-7 Startup
Type X'75' - Queue Posting
Type X'69' - CA-7 Queue Movement

The Type X'69' log record is generated for queue milestones for CPU jobs but not for workstation networks. However, information provided in the log records indicates if there are any networks associated with the job. This indication is included in the reports to alert the user to manually restore those networks to their appropriate status in the CA-7 queues.

The log records contain an image of the CA-7 internal queue records as of the time the log record was written. Type X'69' records are written to the log data set for each of the following milestones:

- When a job initially enters the request queue as a result of an automated or manual scheduling event. The initial entry is only a skeleton record at this time.
- When skeleton records are completely filled in with data from the CA-7 database. This includes all preexecution requirements (for example, overrides, user verification, holds, submit times, and so forth).
- When all requirements for a job have been satisfied and the job is moved from the request queue to the ready queue.
- When JCL in the ready queue is submitted for execution.
- When a job is first initiated by JES and is moved from the ready queue to the active queue.
- When job execution ends, either normally or abnormally, and is moved back to the request queue from the active queue.
- When a job has successfully completed and is moved from the request queue to the prior-run queue. If not successfully completed, a job remains in the request queue until a restart has been manually performed. Once the restart requirement has been satisfied, logging of milestones resumes as for a normal job life.
- When a job is moved back to the request queue as a result of a manually entered REQUEUE top line command or the Q function from a CA-7 QM.1 screen. A job may be requeued from either the ready or active queues.

7.6.1.2 Preserving the Log Data

If the user wishes to generate the recovery aid reports after CA-7 has been restarted, the log data should be unloaded prior to the restart. SASSHIS5, the normal backup program provided, can be used for this purpose. That data can then be used as input to the history reporting facility either before or after CA-7 is restarted. See 8.2, "SASSHIS5" on page 8-4 for more information.

7.6.1.3 Reports Generated

CA-7 log data, requested by a control statement, can be used to report the status of CA-7 jobs at the time of failure. With the log data, the recovery aid program can produce the following reports through the history reporting facility:

SASSRA01 - Last Logged Status of Jobs report
SASSRA02 - Generated Batch Terminal Interface Commands report
SASSRA03 - Simulated LQ Display of Jobs report

An optional data set can also be generated. This data set contains the commands necessary to get the jobs back into the CA-7 request queue. This data set is in standard CA-7 batch terminal interface format.

These reports allow the user to quickly:

- Determine the status of all jobs at the time of failure.
- Restart the necessary jobs after a COLD or FORM start of CA-7.
- Use the command data set so that those jobs that were active at the time of failure can be reentered into the request queue.

The recovery aid reports help in recovery as follows:

- They provide an audit trail of queue contents and generated commands.
- They are an aid in determining whether generated commands should be modified prior to resubmitting them to CA-7.
- They assist in determining the manual recovery required for workstation network considerations for CPU jobs.

See the *CA-7 Reports Guide* for a complete description of these reports.

7.6.1.4 Requesting the Reports

A report type 50 control record is input to the history reporting facility, SASSHIS8, to generate the recovery aid reports. The control record may be input alone or with other history report requests. Optionally, through the control record, the user may request the creation of a data set which can be input to the batch terminal interface. The report type 50 control statement is fully defined in the *CA-7 Reports Guide*.

A start time (and optionally an end time) must be entered into the control record to govern the information to be extracted. The start time must specify a point far enough back in time to ensure that at least one milestone (log record) is included for each job. Otherwise, the job does not appear either in the reports or the generated batch commands. Since a COLD or FORM start clears all of the queues without producing log records, the start time should not be prior to the most recent COLD or FORM start. In the absence of an end time in the control record, log records are extracted until the end-of-file is reached (presumably the point of failure).

Program SASSHR50 prints the recovery aid reports. It selects only the last type X'69' or X'75' record per job written to the log prior to the system failure. This program formats and prints reports SASSRA01 and SASSRA03. If commands were requested in the control record, report SASSRA02 and a file of commands are also produced.

Users of the CA-7 NCF facility must make their NCF node table available to provide node names on the reports. This table should be available through the ddnames STEPLIB or JOBLIB. It could also be defined in the link list.

7.6.1.5 Recovery Aid Procedures

It is impossible to document all possible situations which might occur necessitating the use of the recovery aid output. The following is intended only as a general guideline.

1. Establish procedures for management of CA-7 log data so that if a system or CA-7 failure should occur, all log data is saved. These procedures must ensure saving log files before a cold type (COLD, FORM, MOVQ) start is attempted. The SASSHIS5 program/job should be used to save the data and accumulate it into the log history file.
2. Implement procedures for also saving master station output. This output may be gathered by running the SASSHIS8 browse data set report. See the *CA-7 Reports Guide* for a description. Although the recovery aid output is probably more useful than the master station output, any jobs which had workstation networks in-process at the time of the failure may require this output to determine their status.
3. After ensuring that log and master station output have been saved, try to restart CA-7 with either WARM or ERST (emergency restart).
4. If CA-7 comes up successfully, it is unnecessary to continue this procedure.
5. Otherwise, produce the recovery aid output, using the history reporting job and the log data saved and accumulated in Step 1.

Specify a start date/time that is at least prior to the last schedule scan which scheduled jobs that were still in the queues. Do not specify a start date/time which is earlier than the last COLD or FORM start of CA-7. A date/time value earlier than one of those types of CA-7 restarts causes unnecessary records to be selected and causes the recovery aid report generation to take longer than it would otherwise.

Specify an ending date/time that is later than the system failure. Any value later than the last log record on the log history file works since that causes all records up to the end of the file, the last log record written before the failure, to be considered.

If the batch terminal interface command data set commands is desired, also specify either DEMAND or DEMANDH in the report type 50 control record. DEMANDH is the recommended value.

6. COLD start CA-7. Step 6 may be done before, after or concurrently with Step 5. It may be desirable to specify RUNOPT=NSTA instead of SCAN to deactivate schedule scan for the time being. A FORM start may be necessary if one of the queues has been destroyed.

7. Using the recovery aid reports, research the status of all CPU jobs that were in the request, ready, or active queues. Once Step 6 has been completed, all CA-7 inquiries, forecasts, and so forth, are also available for further assistance in this process.
8. Perform any necessary file maintenance, JCL overrides, and so forth. If CA-11 was being used in all of the jobs, file maintenance is probably unnecessary.
9. If generated in Step 5, edit the commands data set as necessary to accomplish whatever action was determined in Step 7 to be correct for each job.

Be sure to provide the proper operator ID in the /LOGON command.

10. Once the commands data set is in order, run a batch terminal interface job using the commands data set as the input data.

If only a few jobs were in the queues at the time of the failure, it may be easier to issue online commands to get the jobs back into the queues. The user must decide which technique is most appropriate for the number of jobs involved.

If CA-7 was activated without schedule scan, use the SSCAN command and an appropriate PERSTART time to reactivate schedule scan.

If USERID security was in use, some of the DEMANDED jobs may not be scheduled correctly depending on the operator ID used by the batch terminal interface run.

11. The NETWORKS field on the SASSRA01-01 Last Logged Status of Jobs report from Step 5 and the master station output from Step 2 can be of assistance in determining which workstation networks, if any, were in process and what their status was at the time of the failure.

Input networks have to be manually rescheduled and reinstated to the appropriate status.

Output networks are automatically scheduled back into the queues by the generated commands for the associated CPU jobs; however, their status has to be reinstated manually if the network had progressed anywhere beyond the LOGIN of the first workstation.

Do not overlook jobs that may have made it to the prior-run queue before the failure but still had output networks in-process. (No commands are generated for jobs that had made it to the prior-run queue.) Any such networks have to be manually rescheduled and reinstated just as input networks must be:

```
//jobname JOB .....
//stepname EXEC CA7LOG,PG=SASSHIS8
//COMMANDS DD DSN=user.ca7.recovery.commands,
//            DISP=(,CATLG,DELETE),UNIT=SYSDA,
//            SPACE=(CYL,(5,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=nnnnn)
//UCC7ARCH DD DUMMY
//UCC7HIST DD DSN=user.ca7.loghist(0),DISP=OLD
//SASSRA01 DD SYSOUT=A
//SASSRA02 DD SYSOUT=A
//SASSRA03 DD SYSOUT=A
//SYSIN DD *
50 RECOVERY yydddhmm DEMANDH
/*
```

Figure 7-5. Recovery Aid JCL

Note: See also the SASSHIS8 discussion in the *CA-7 Reports Guide*. The CA7LOG procedure is generated during installation and may have a different name at your site.

Chapter 8. Log and History Data Set Management

This chapter discusses the facilities available for managing the Log data which is produced by CA-7 and used to produce the history and recovery aid reports.

8.1 Log Data Set Management

For processing the CA-7 log data sets, you must define separate jobs in the database for dumping each data set. When one log becomes full, recording activity automatically switches to the alternate log. In addition to switching from one log to the other, CA-7 automatically schedules the appropriate job for dumping the full log.

You must define the jobs to be used for dumping the log in the CA-7 database. The job name of the job to dump the primary log must have an eighth character of P, for primary. The job name for the secondary log dump job must have the same first 7 characters as the primary log dump job. An eighth character of S, for secondary, is required.

The log dump job must be identified in the CA-7 initialization file. To do this, specify the job name of the log dump job on the DBASE statement. If two logs are defined, the job name of the primary log dump job must be specified.

The JCL for the log dump jobs (and the appropriate initialization file statements) is generated during the SYSGEN of CA-7. During batch execution of CA-7 at installation time, log dump jobs were added to the database. The procedure, CA7LOG, was added to a PROCLIB in the N020 installation job.

These jobs may also be used to unload log data, following a system failure and prior to restarting CA-7, for input to the recovery aid procedure. Primary Log Dump JCL Example on page 8-3 shows sample JCL for a log dump job.

The secondary log dump job, shown as Secondary Log Dump JCL Example on page 8-3 is a duplicate of the primary log dump job with the following two exceptions:

- The JOB name must end with an S (S for secondary).
- The DD statement for LOGIN should reference the secondary data set.

The LOGIN data set name in each job must refer to the appropriate log data set as it is cataloged on the OS catalog.

The LOGIN, HISTIN, and HISTOUT data set names and the job names may be changed to meet installation standards.

It is recommended that the data set to which the log is dumped be a GDG with at least three entries.

```

//userlogP JOB .....
//* CA-7 SYSGEN JOB
//STEP1 EXEC CA7LOG
//LOGIN DD DISP=SHR,DSN=user.primary.log.dataset,DCB=RECFM=VB
//HISTIN DD DISP=SHR,DSN=user.logtape(+0)
//HISTOUT DD DSN=user.logtape(+1),DISP=(NEW,CATLG,DELETE),
//          UNIT=tape,LABEL=EXPDT=990000,
//          DCB=(RECFM=VB,LRECL=2100,BLKSIZE=32760)

```

Figure 8-1. Primary Log Dump JCL Example

```

//userlogS JOB .....
//* CA-7 SYSGEN JOB
//STEP1 EXEC CA7LOG
//LOGIN DD DISP=SHR,DSN=user.secondary.log.dataset,DCB=RECFM=VB
//HISTIN DD DISP=SHR,DSN=user.logtape(+0)
//HISTOUT DD DSN=user.logtape(+1),DISP=(NEW,CATLG,DELETE),
//          UNIT=tape,LABEL=EXPDT=990000,
//          DCB=(RECFM=VB,LRECL=2100,BLKSIZE=32760)

```

Figure 8-2. Secondary Log Dump JCL Example

8.2 SASSHIS5

Name: History Management program

Purpose: Extracts and manages the CA-7 log records produced by the CA-7 system.

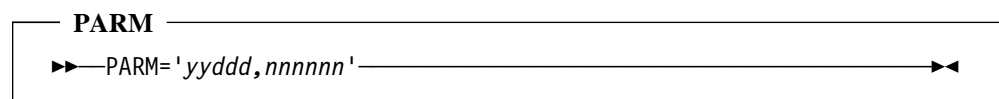
Input: CA-7 log data file
CA-7 log history file

Output: CA-7 log archives file (optional)
CA-7 log history file
History Management report

Narrative: SASSHIS5 collects CA-7 log records from the CA-7 primary or secondary log files and stores them in a control sequence on the CA-7 log history file. The resulting report reflects the current contents of the log history file as record counts within the date and time-of-day ranges in which the records were generated.

At the user's option, SASSHIS5 can also remove log records from the log history file and place them on the CA-7 log archives file. This transfer can occur either while the log records are being extracted from the log files or as a separate operation by making the DD for the log files a DD DUMMY. See Figure 8-3 on page 8-5 for a flowchart of the history management facility SASSHIS5.

PARM operand: The optional PARM operand of the EXEC statement allows two fixed-length positional keywords as follows:



Where:

yyddd

Is a Julian date specifying the ending date of the period to be placed onto the CA-7 log archives file. Five blanks cause no records to be archived. The value TODAY may be used to specify the current date. There is no default for this parameter.

nnnnnn

Is a 6-digit number, with leading zeros, representing the amount of memory available to the internal sort. Must be 6 blanks if not used. Maximum available memory is the default.

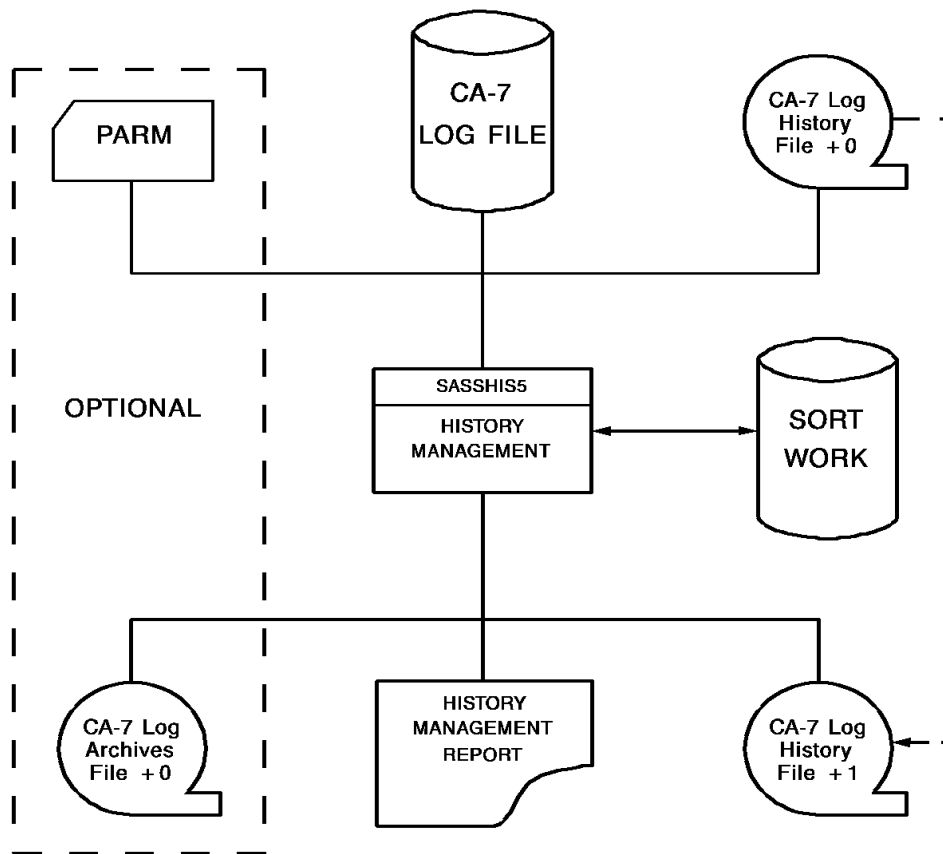


Figure 8-3. SASSHIS5 Flowchart

8.2.1 File Descriptions

File name	DDNAME	Description
Log file	LOGIN	The CA-7 log file produced by CA-7. This is the primary source of information for SASSHIS5.
Log history	HISTIN	The generation data set containing the accumulated history available for analysis.
Log archives	ARCHOUT	The generation data set containing the accumulated history purged from the CA-7 log history and saved for analysis. This can be a MOD file.
Log history	HISTOUT	The generation data set containing accumulated history available for analysis.

JCL: Following is an example of the log archive JCL (also see the N535 installation job).

```
//jobname JOB .....
//HISTARC EXEC CA7LOG,PA='yyddd,nnnnnn'
//LOGIN DD DUMMY,DCB=BLKSIZE=24
//HISTIN DD DISP=SHR,DSN=user.logtape(+0)
//HISTOUT DD DSN=user.logtape(+1),DISP=(NEW,CATLG,DELETE),
// UNIT=tape,LABEL=EXPDT=990000,
// DCB=(RECFM=VB,LRECL=2100,BLKSIZE=32760)
//ARCHOUT DD DSN=user.logarch(+0),DISP=(MOD,CATLG,KEEP),
// UNIT=tape,LABEL=EXPDT=990000,
// DCB=(RECFM=VB,LRECL=2100,BLKSIZE=32760)
```

8.2.2 Report Descriptions

Report Name	DDNAME	Description
History Management	CNTLREPT	The report reflecting current activity in SASSHIS5 and the contents of log history and log archives.

Figure 8-4 is a History Management report sample.

HIS5-01 PARM=yy215,200000													
SASSHIS5		C A - 7 -		HISTORY MANAGEMENT				08/02/yy 11:03		PAGE 002			
LOG DATA CATEGORY		EXTRACTED DATE TIME		P/ S	STARTING DATE TIME		ENDING DATE TIME		RECORD COUNT	D S	ARCHIVES DATE	EXTRACTED TIME	DUPLICATE COUNT
HISTORY EXTRACT		02/12/yy 10:31 S			02/12/yy 10:28		02/12/yy 10:28		00000001		05/17/yy 16:15		
		02/19/yy 09:15 S			02/18/yy 10:42		02/18/yy 10:42		00000001		05/17/yy 16:15		
		02/19/yy 16:04 S			02/19/yy 14:22		02/19/yy 14:22		00000001		05/17/yy 16:15		
		03/01/yy 10:12 S			02/29/yy 14:23		02/29/yy 14:23		00000001		05/17/yy 16:15		
		05/14/yy 08:54 S			05/13/yy 15:12		05/13/yy 15:12		00000001		05/17/yy 16:15		
HISTORY EXTRACT		05/14/yy 08:54 S			05/17/yy 14:03		05/17/yy 16:48		00002456		05/18/yy 09:11		
		05/18/yy 09:03 P			05/17/yy 16:48		05/17/yy 19:44		00000390		05/18/yy 09:11		
		05/18/yy 09:03 P			05/18/yy 09:03		05/18/yy 09:03		00000001		05/18/yy 15:49		
		05/18/yy 15:09 S			05/18/yy 09:04		05/18/yy 15:08		00005036		05/18/yy 15:49		
		05/18/yy 15:09 P			05/18/yy 15:08		05/18/yy 20:47		00001166		05/25/yy 10:50		
HISTORY EXTRACT		05/19/yy 08:58 P			05/18/yy 20:48		05/18/yy 20:48		00000001		05/25/yy 10:50		
		05/19/yy 08:58 S			05/18/yy 20:49		05/19/yy 01:36		00001162		05/25/yy 10:50		
		05/19/yy 08:58 P			05/19/yy 08:57		05/19/yy 08:57		00000001		05/25/yy 10:50		
		05/19/yy 12:10 P			05/19/yy 08:58		05/19/yy 12:08		00002054		05/25/yy 10:50		
		05/19/yy 16:29 P			05/19/yy 12:09		05/19/yy 12:09		00000001		05/25/yy 10:50		
HISTORY EXTRACT		05/19/yy 16:29 S			05/19/yy 12:10		05/19/yy 14:33		00000517		05/25/yy 10:50		
		05/19/yy 16:27 P			05/19/yy 16:27		05/19/yy 16:27		00000001		05/25/yy 10:50		
		05/19/yy 21:06 P			05/19/yy 16:28		05/19/yy 21:05		00001623		05/25/yy 10:50		
		05/20/yy 12:39 S			05/19/yy 21:05		05/20/yy 12:36		00002991		05/25/yy 10:50		
		05/20/yy 12:38 P			05/20/yy 12:38		05/20/yy 12:38		00000022		05/25/yy 10:50		
HISTORY EXTRACT		05/20/yy 16:22 P			05/20/yy 12:38		05/20/yy 16:20		00002118		05/25/yy 10:50		
		05/23/yy 09:02 P			05/20/yy 16:21		05/20/yy 16:21		00000001		05/25/yy 10:50		
		05/23/yy 09:02 S			05/20/yy 16:22		05/20/yy 20:51		00000380		05/25/yy 10:50		
		05/23/yy 09:01 P			05/23/yy 09:01		05/23/yy 09:01		00000001		05/25/yy 10:50		
		05/23/yy 11:39 P			05/23/yy 09:01		05/23/yy 11:38		00002511		05/25/yy 10:50		
HISTORY EXTRACT		05/23/yy 15:14 S			05/23/yy 11:38		05/23/yy 15:11		00010197		05/25/yy 10:50		
		05/23/yy 15:34 P			05/23/yy 15:34		05/23/yy 15:34		00000002		05/25/yy 10:50		
		05/23/yy 15:38 P			05/23/yy 15:34		05/23/yy 15:35		00000129		05/25/yy 10:50		
		05/23/yy 15:50 P			05/23/yy 15:44		05/23/yy 15:44		00000001		05/25/yy 10:50		
		05/23/yy 15:50 S			05/23/yy 15:46		05/23/yy 15:47		00000199		05/25/yy 10:50		
HISTORY EXTRACT		05/23/yy 17:33 P			05/23/yy 16:42		05/23/yy 17:32		00003092		05/25/yy 10:50		
		05/24/yy 09:05 S			05/23/yy 17:32		05/24/yy 01:28		00002043		05/25/yy 10:50		
		05/24/yy 09:04 P			05/24/yy 09:04		05/24/yy 09:04		00000022		05/25/yy 10:50		
		05/24/yy 10:22 P			05/24/yy 09:04		05/24/yy 10:21		00002873		05/25/yy 10:50		
		05/24/yy 15:45 S			05/24/yy 10:21		05/24/yy 15:44		00012954		05/25/yy 10:50		
HISTORY EXTRACT		05/24/yy 18:17 S			05/24/yy 15:44		05/24/yy 15:44		00000001		05/25/yy 10:50		
		05/24/yy 18:17 P			05/24/yy 15:44		05/24/yy 18:15		00002890		05/25/yy 10:50		
		05/25/yy 10:22 P			05/24/yy 18:16		05/24/yy 18:16		00000001		05/25/yy 10:50		
		05/25/yy 10:22 S			05/24/yy 18:17		05/25/yy 00:13		00000468		05/25/yy 10:50		
		05/25/yy 10:20 P			05/25/yy 10:20		05/25/yy 10:20		00000001		05/25/yy 10:50		
		05/25/yy 10:20 P			05/25/yy 10:20		05/25/yy 11:34		00002996		05/27/yy 08:04		
		05/25/yy 15:58 S			05/25/yy 11:34		05/25/yy 15:55		00013603		05/27/yy 08:04		
		05/25/yy 19:34 P			05/25/yy 15:55		05/25/yy 19:32		00002381		05/27/yy 08:04		
		05/26/yy 12:06 P			05/25/yy 19:33		05/25/yy 19:33		00000001		05/27/yy 08:04		
		05/26/yy 12:06 S			05/25/yy 19:34		05/26/yy 12:03		00007848		05/27/yy 08:04		
		05/26/yy 16:37 P			05/26/yy 12:03		05/26/yy 14:14		00003101		05/27/yy 08:04		
		05/27/yy 05:24 P			05/26/yy 14:14		05/26/yy 14:14		00000018		05/27/yy 08:04		
		05/27/yy 05:24 S			05/26/yy 14:14		05/27/yy 05:22		00008226		05/27/yy 08:04		
		05/27/yy 05:24 S			05/27/yy 05:23		05/27/yy 05:23		00000001		06/01/yy 10:11		
		05/27/yy 09:39 P			05/27/yy 05:24		05/27/yy 09:38		00001704		06/01/yy 10:11		

Figure 8-4. History Management Report Sample

Item	Description
------	-------------

LOG DATA CATEGORY

The disposition of the CA-7 log records are represented as follows:

Category	Description
----------	-------------

EXTRACTED DATE TIME

Data was removed from the log history file and placed onto the log archives file.

LOG EXTRACT

Data was extracted from the log history files and placed onto the log history file.

Spaces

Data is a continuation of the previous nonblank category. Results when a swap occurs between the primary and secondary log files due to an overflow.

EXTRACTED DATE TIME

The date and time log records were actually extracted and placed onto the log history file.

P/S

Log file data was extracted from primary or secondary.

STARTING DATE TIME

The date and time the first record of this batch was written to the log file.

ENDING DATE TIME

The date and time the last record of this batch was written to the log file.

RECORD COUNT

The number of log records extracted for the represented CA-7 session.

ARCHIVES EXTRACTED DATE TIME

The date and time the log records represented were removed from log history and placed onto log archives. Blank for LOG DATA CATEGORY value of LOG EXTRACT.

DUPLICATE COUNT

Number of duplicated records which were bypassed.

Note: The History Management report also displays the options used in the PARM and any associated error messages. See the *CA-7 Message Guide* for detailed descriptions of these error messages.

8.3 SASSHIS6

Name: History Purge program

Purpose: Purges unwanted CA-7 log records from the CA-7 log archives file and log history file.

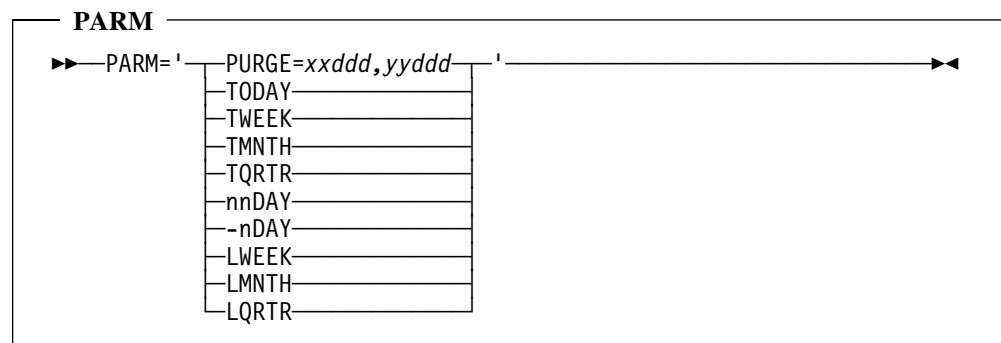
Input: CA-7 log history file
CA-7 log archives file

Output: CA-7 log history file
CA-7 log archives file
Archives Purge report

Required parameter: PARM='parameter'

Narrative: SASSHIS6 deletes unwanted CA-7 log records from the log history and log archives files while keeping the internal control data between the files synchronized. SASSHIS6 should run whenever the older CA-7 log records are no longer needed for analysis. These records are permanently deleted from the system and thereby prevent the log files from becoming too large. The log files created from SASSHIS6 are the new files used in subsequent history reporting activities.

PARM operand: The required PARM operand of the EXEC statement must contain one of the following purge parameters.



Where:

PURGE

Identifies the from and through dates defining the period (or range) to be purged.

xxddd

Is a Julian date specifying the start of the period to be deleted from the files (from date). No default.

yyddd

Is a Julian date specifying the end of the period to be deleted from the files (thru date). No default.

Literals for to-date purge control and their meanings are as follows:

TODAY

Indicates delete all data with the current date.

TWEEK

Indicates delete all data produced this calendar week, last Sunday's date through today.

TMNTH

Indicates delete all data produced this calendar month, beginning with the date of the first day of the current month through today.

TQTR

Indicates delete all data produced this quarter, beginning with the date of the first day of the month two months ago, plus the current month through today. (Not calendar quarter.)

Literals for prior periods (whose end times have already passed) and their meanings are as follows:

nnDAY

Indicates delete the previous nn days through current date/time.

-nDAY

Indicates data from the previous n 24-hour periods is to be deleted. This generates a control statement with a beginning time of 0000 and an ending time of 2400, encompassing the number of days specified by n (where ending date/time is yesterday at midnight).

LWEEK

Indicates delete data from the previous Sunday through Saturday.

LMNTH

Indicates delete the previous calendar month's data.

LQTR

Indicates delete the previous three consecutive months' data.

Figure 8-5 illustrates the flow of the SASSHIS6 program.

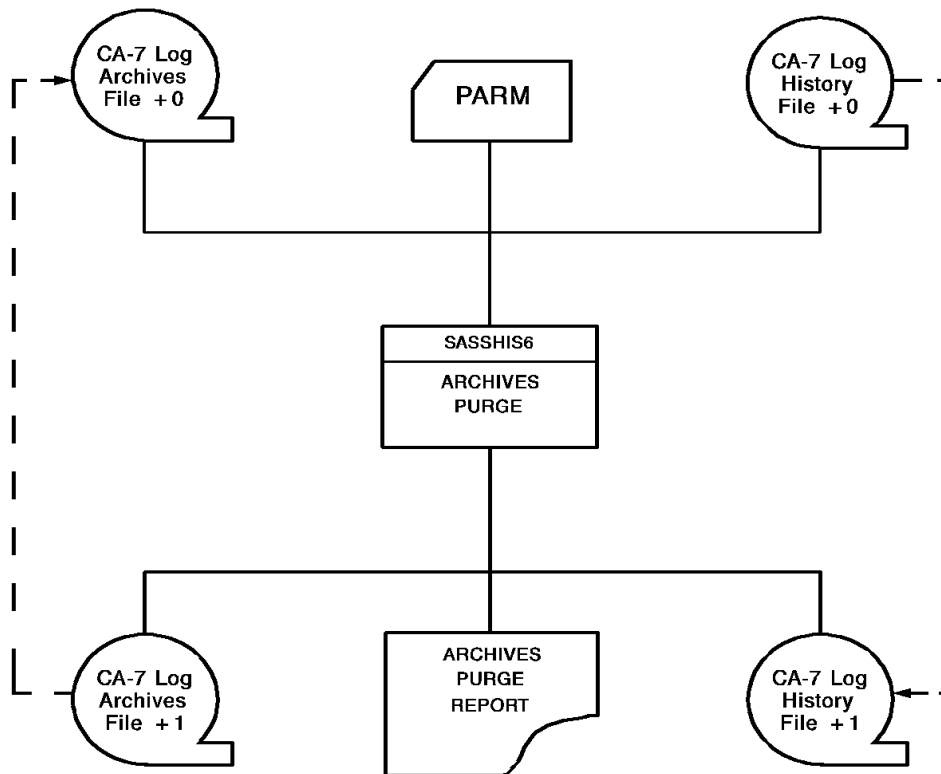


Figure 8-5. SASSHIS6 Flowchart

8.3.1 File Descriptions

File Name	DDNAME	Description
Log History	HISTIN	The generation data set containing the accumulated history available for analysis and deletion.
Log Archives	ARCHIN	The generation data set containing the accumulated history previously transferred from the log history file and available for deletion.
Log History	HISTOUT	The generation data set containing accumulated history available for analysis after the specified history data has been deleted.
Log Archives	ARCHOUT	The generation data set containing the accumulated history (transferred originally from the log history file) after the specified history data has been deleted.

JCL: Following is an example of the log purge JCL (also see the N540 installation job).

```
//jobname JOB ...
//PURGE EXEC CA7LOG,PG=SASSHIS6,PA='PURGE=xxddd,yyddd'
//HISTIN DD DISP=SHR,DSN=user.logtape(+0)
//HISTOUT DD DSN=user.logtape(+1),DISP=(NEW,CATLG,DELETE),
//          UNIT=TAPE,LABEL=EXPDT=990000,
//          DCB=(RECFM=VB,LRECL=2100,BLKSIZE=32760)
//ARCHIN DD DISP=SHR,DSN=user.logarch(+0)
//ARCHOUT DD DSN=user.logarch(+1),
//           DISP=(NEW,CATLG,DELETE),
//           UNIT=TAPE,LABEL=EXPDT=990000,
//           DCB=(RECFM=VB,LRECL=2100,BLKSIZE=32760)
//DEL EXEC PGM=IEFBR14,COND=(4,NE)
//ARCHIVES DD DSN=user.logarch(+0),DISP=(OLD,DELETE)
```

Note: A condition code of 4 from SASSHIS6 indicates that a new archive file was not created because no data was being purged from the ARCHIN data set.

A condition code of 8 from SASSHIS6 indicates that a PARM error was detected.

8.3.2 Report Descriptions

Report Name	DDNAME	Description
Archives Purge	SYSLIST	The report reflecting current activity in SASSHIS6 and the new contents of log history and log archives.

Figure 8-6 is a sample of the Archives Purge Report.

SASSHIS6		C A - 7 - ARCHIVES PURGE					07/25/yy 10:39		PAGE 001	
LOG DATA CATEGORY	EXTRACTED DATE TIME		STARTING DATE TIME		ENDING DATE TIME		RECORD COUNT	ARCHIVE DATE TIME	EXTRACT DATE TIME	PURGED DATE TIME
CURRENT ARCHIVES	02/19/yy	09:15	02/18/yy	10:42	02/18/yy	10:42	00000001	05/17/yy	16:15	
	02/19/yy	16:04	02/19/yy	14:22	02/19/yy	14:22	00000001	05/17/yy	16:15	
	03/01/yy	10:12	02/29/yy	14:23	02/29/yy	14:23	00000001	05/17/yy	16:15	
	05/14/yy	08:54	05/13/yy	15:12	05/13/yy	15:12	00000001	05/17/yy	16:15	
	05/14/yy	08:54	05/17/yy	14:03	05/17/yy	16:48	00002456	05/18/yy	09:11	
	05/18/yy	09:03	05/17/yy	16:48	05/17/yy	19:44	00000390	05/18/yy	09:11	
	05/18/yy	09:03	05/18/yy	09:03	05/18/yy	09:03	00000001	05/18/yy	15:49	
	05/18/yy	15:09	05/18/yy	09:04	05/18/yy	15:08	00005036	05/18/yy	15:49	
	05/18/yy	15:09	05/18/yy	15:08	05/18/yy	20:47	00001166	05/25/yy	10:50	
	05/19/yy	08:58	05/18/yy	20:48	05/18/yy	20:48	00000001	05/25/yy	10:50	
CURRENT ARCHIVES	05/19/yy	08:58	05/18/yy	20:49	05/19/yy	01:36	00001162	05/25/yy	10:50	
	05/19/yy	08:58	05/19/yy	08:57	05/19/yy	08:57	00000001	05/25/yy	10:50	
CURRENT ARCHIVES	05/19/yy	12:10	05/19/yy	08:58	05/19/yy	12:08	00002054	05/25/yy	10:50	
	05/19/yy	16:29	05/19/yy	12:09	05/19/yy	12:09	00000001	05/25/yy	10:50	
	05/19/yy	16:29	05/19/yy	12:10	05/19/yy	14:33	00000517	05/25/yy	10:50	
	05/19/yy	16:27	05/19/yy	16:27	05/19/yy	16:27	00000001	05/25/yy	10:50	
CURRENT ARCHIVES	05/19/yy	21:06	05/19/yy	16:28	05/19/yy	21:05	00001623	05/25/yy	10:50	
	05/20/yy	12:39	05/19/yy	21:05	05/20/yy	12:36	00002991	05/25/yy	10:50	
	05/20/yy	12:38	05/20/yy	12:38	05/20/yy	12:38	00000022	05/25/yy	10:50	
	05/20/yy	16:22	05/20/yy	12:38	05/20/yy	16:20	00002118	05/25/yy	10:50	
	05/23/yy	09:02	05/20/yy	16:21	05/20/yy	16:21	00000001	05/25/yy	10:50	
	05/23/yy	09:02	05/20/yy	16:22	05/20/yy	20:51	00000380	05/25/yy	10:50	

Figure 8-6. Archives Purge Report Sample

Item Description

LOG DATA CATEGORY

The disposition of the CA-7 log records are represented as follows:

Category Description

CURRENT HISTORY

Data currently on the log history file.

HISTORY PURGE

Data deleted from the log history file.

CURRENT ARCHIVES

Data currently on the log archives file.

ARCHIVES PURGE

Data deleted from the log archives file.

EXTRACTED DATE TIME

The date and time log records were actually extracted.

STARTING DATE TIME

The date and time the first record of this batch was written to the log file.

ENDING DATE TIME

The date and time the last record of this batch was written to the log file.

RECORD COUNT

The number of log records extracted for the represented CA-7 session.

ARCHIVE EXTRACT DATE TIME

The date and time the log records represented were extracted from either log history or log archives.

PURGED DATE TIME

The date and time log records represented were purged from either log history or log archives.

Note: The Archives Purge Report also displays the options used in the PARM parameter and any associated error messages. See the *CA-7 Message Guide* for detailed descriptions of these error messages.

Chapter 9. User Exits and Modifications

This chapter discusses coding user exits, online and batch exits, changing graph definitions, adding counters to new reports, and other system modification techniques.

9.1 CA-7 User Modifications Under SMP

A list of all CA-7 user modifications available under SMP is provided in the `$$INDEX` member of the CA-7 SAMPJCL file. USERMOD member names are prefixed with UL233.

9.2 Coding User Exits

User exits provide the capability to tailor CA-7 to meet the special needs of an installation. See the `$$INDEX` member in the CA-7 SAMPJCL file (members prefixed by `UL233`) for the appropriate user exit `USERMODs`.

Extreme care should be taken to observe the conventions and restrictions regarding the use and coding of user exits as outlined in this chapter. Failure to do so could result in serious degradation of CA-7 system functions.

There are two classes of user exits: online and standard exits. In the online exit environment, CA-7 online services are available and must be used for such things as entry and exit linkage and requests for storage. In the standard exit environment, CA-7 online services may not be available. Whether an exit is considered online or standard imposes several important restrictions that must be observed in exit code. Because online exits require the use of CA-7 system services, they cannot be tested outside the CA-7 online environment.

The following table classifies the CA-7 user exits as either online or standard.

Exit name	Exit type
Exit invoked by SASBK00	standard
SASSBCX1	standard
Exit invoked by SASPM00 (Problem Management Interface exit)	standard
SASSXXBT	standard
SASSXXFF	online
SASSXXLX	online
SASSXXQB	online
SASSXXQD	online
SASSXX01	online
SASSXX02	online
SASSXX03	online
SASSXX04	online
SASSXX05	online
SASSXX07	online
SASSXX08	online
SASSXX09	online
SASSXX10	online
SASSXX11	online
SASSXX12	standard
SASSXX13	online
SASSXX14	online
SASSXX15	online
SASSXX16	online

9.2.1 Rules for Coding Online Exits

The following rules apply to coding online user exits. In certain cases, additional restrictions may apply; see the description of the exit you wish to code for further information.

1. Must be written in assembler language.
2. Must include the following macros:
 - EXITPARM (to map exit parameters)
 - SASSEQU (declares equates required by CA-7 macros, also provides register equates)
 - UCC7SVT (required by many CA-7 services)
 - SASSVRSN can be used for entry
 - In general, must use SEXIT to return
3. Should be coded reentrant and serially reusable unless otherwise noted.
4. Any activity which has the potential to degrade CA-7 system functioning should be avoided. OS/390 services that must not be requested include but are not limited to: WAIT, STIMER, and SVC 99. Services which could cause a wait condition or an STIMER to be issued could adversely affect system performance and must not be used. Because many online exits are invoked directly from critical CA-7 functions it is important that exit routines be of short duration.
5. An APPLCTN statement identifying the exit must be included in the CA-7 initialization file. If the size of the exit module exceeds 32K, then ATTR=PERM must be specified on the APPLCTN statement.
6. The entry point must be the beginning of the load module.

9.2.2 Register Descriptions for Online Exits

The following table summarizes the register conventions for the online user exits. See individual exits for specific register contents.

Table 9-1. Register Conventions for Online User Exits	
Register	Convention
0	Destroyed
1	Destroyed
2	Saved across CA-7 system requests
3	Saved across CA-7 system requests
4	Saved across CA-7 system requests
5	Saved across CA-7 system requests
6	Saved across CA-7 system requests
7	Saved across CA-7 system requests
8	Saved across CA-7 system requests
9	Saved across CA-7 system requests (secondary base register)
10	Saved across CA-7 system requests (primary base register)
11	Points to the SCT - must not be modified - mapped by macro SCTENTRY
12	Points to the SVT - must not be modified - mapped by macro UCC7SVT
13	Destroyed - does not point to a save area unless otherwise noted
14	Saved across CA-7 system requests
15	Destroyed - but upon entry contains the entry address

9.2.3 Rules for Coding Standard Exits

The following rules apply to coding STANDARD user exits. In certain cases additional restrictions may apply, see the description of the exit you wish to code for further information.

1. Must be written in assembler language.
2. Should include the EXITPARM macro to map exit parameters.
3. Should be coded reentrant and serially reusable unless otherwise noted.

9.2.4 Testing User Exits

During testing, it may be necessary to refresh CA-7 exit modules. In certain cases, this may be accomplished using the /RELINK command.

If the detailed description of the exit indicates that /RELINK may be used, then /RELINK may refresh the module if it is not marked PERM or RESD. If the module is marked PERM or RESD, then the CA-7 address space must be recycled to refresh the module.

See the *CA-7 Commands Guide* for information on /RELINK.

9.3 Online Exit Descriptions

9.3.1 Job Name Verification

Application Name: SASSXX01 SAMPJCL: UL23317

Exit Purpose: To validate a job name. Must be reentrant.

Caller: Control is passed to the exit from SASSSM2A. This exit may be /RELINKed.

Module Attributes: This exit must use reentrant coding.

Input: R2 = Address of parameter list

 +0 = Address of 8-byte area containing job name

Output: R15 = 0 Job name is valid

 R15 Not equal to 0 Invalid job name

Note: Code EXITPARM EXIT=SASSXX01 to map input.

CA-7 validates that the job name is valid for OS before the exit receives control.

If a nonzero return code is passed back in R15, the user receives a message as indicated below:

This exit is invoked by and for the following:

- The DB.1 screen ADD function. If error returned, message is:
SM20-05 INVALID OR OMITTED JOB DATA
- The RUN(H) function on the DB.7 screen. If error returned, message is:
SM50-05 INVALID OR OMITTED JOBCARD DATA
- Commands DEMAND(H), RUN(H), and LOAD(H) for a job not in the database. If error returned, message is:
SPO7-01 JOBNAME - INVALID
- Command ADDRQ to validate the DEPJOB value. If error returned, message is:
SPO4-02 ADDRQ UNSUCCESSFUL DEPJOB -- INVALID

9.3.2 JCL Submission

Application Name: SASSXX02 SAMPJCL: UL23318

Exit Purpose: Used to add, delete, or change JCL statements or control statements immediately preceding job submission. Changes made to JCL by this exit are not reflected in the queue JCL.

Caller: Control is passed to the exit from SASSSSM0. This exit may be /RELINKed.

Module Attributes: Must be coded as reusable, but not necessarily reentrant.

Input: R2 = Address of parameter list

- +00 = Address of 80-byte area containing the JCL statement
- +04 = Address of job's queue record (mapped by the CA-7 macro - JQREC)
- +08 = Fullword containing an entry code
 - 0 - not end-of-data call
 - 1 - end-of-data call. Note that SASSXX02 will not be called at end-of-data for job unless CA-7 has been instructed to do so by flagging the communication area. On the end-of-data call, the address at +00 will point to a comment statement.
- +12 Four byte communication area used by exit to set calling options.
 - X'80' instructs CA-7 to call SASSXX02 at end of data. This flag must be set for each job that is to receive an end-of-data call.
- +16 Fullword reserved for use by exit. The value in this area will not be modified by CA-7. It is initialized to zeros for each job processed.

Output: R15 = Return code

- 00 = Write the statement and continue processing (the JCL statement may have been modified).
- 01 = Delete this statement from the JCL.
- 02 = Write the statement and return to the exit prior to fetching additional JCL (used to add statements).

The job name and columns 67 through 71 in the JOB statement must not be changed. The only areas in the parameter list that may be changed by the exit are the communication area and the fullword reserved for client use. The exit must NOT modify any other areas in the parameter list.

Note: Use EXITPARM EXIT=SASSXX02 to map input parameters.

There is a 20-byte user field in the JQREC DSECT that can be modified by the user. This field is named JQUSER. If modified, care must be exercised that only the specified 20 bytes are used. The JQUSER field is copied to the JQUSER field of any triggered job. The JQUSER field is initialized (hex zeros) only when the job is initially placed into the REQQ by schedule scan and DEMAND. This field is propagated for triggered jobs. This field is not initialized for restarted jobs.

9.3.3 Utility Function Security Checking

Application Name: SASSXX03 SAMPJCL: UL23319

Exit Purpose: Used to reject utility requests. Must be reentrant. This exit is called by the utility commands specified below:

Caller: This exit receives control through an SLINK from one of the following modules:

SASSUTL3
SASSUTL4
SASSUTL5
SASSUTL6

This exit may be /RELINKed.

Module Attributes: This module must use reentrant coding.

Input: R2 = Address of parameter list. This list varies with the function. The first word in the parameter list points to an 8-byte area containing the function to be performed. (The command that was entered.) Other offsets in the parameter list are explained in the following table:

AL or ALC	(Allocate or allocate and catalog)
+4 =	Address of 44-byte DSN
+8 =	Address of 6-byte volume serial number
BLDG	(Build GDG index)
+4 =	Address of 44-byte index name
CAT	(Catalog)
+4 =	Address of 44-byte DSN
+8 =	Address of 30-byte area containing one to five volume serial numbers
+12 =	Address of 6-byte volume serial number if CVOL specified, zero if CVOL was not specified
CONN	(Connect index)
+4 =	Address of 8-byte index name
+8 =	Address of 6-byte volume serial number
DCONN	(Disconnect index)
+4 =	Address of 8-byte index name
DLTX	(Delete index)
+4 =	Address of 44-byte index name

+8 =	Not used
+12 =	Address of 6-byte volume serial number if CVOL specified, zero if CVOL was not specified
RENAME	(Rename data set)
+4 =	Address of 44-byte DSN
+8 =	Address of 6-byte volume serial number
+12 =	Address of last 96 bytes of format 1 DSCB
+16 =	Address of 44-byte new name
SCRATCH SCRATCHP	(Scratch a data set or scratch a protected data set)
+4 =	Address of 44-byte DSN
+8 =	Address of 6-byte volume serial number
+12 =	Address of last 96 bytes of format 1 DSCB
UNC	(Uncatalog a data set)
+4 =	Address of 44-byte DSN
+8 =	Address of 6-byte volume serial number

Note: In all cases, at +24 in the parameter list is a word that contains a value set on the SASSXXLX exit or zeros. Use EXITPARM EXIT=SASSXX03 to map these parameters.

Output: R15 = 0 Process the request
R15 Not equal to 0 Reject the request

9.3.4 ENQ/RESERVE

Application Name: SASSXX04 SAMPJCL: UL23320

Exit Purpose: When a SAVE, REPL, RENAME, or DELETE is done from the DB.7 screen, CA-7 issues an ENQ or RESERVE to maintain integrity of the data set. In the case of online systems, it may be desirable to change the QNAME or completely bypass the ENQ/RESERVE with this exit.

Note: This exit does not get control if the dynamic allocation feature is used. (See PERFORM keyword on INIT statement in the initialization file.)

Caller: This exit receives control through an SLINK from module SASSSM5C. This exit may be /RELINKed.

Module Attributes: This exit must use reentrant coding.

Input: R2 = Address of parameter list

- +0 = Address of QNAME (SYSDSN)
- +4 = Address of data set name
- +8 = Address of volume serial number
- +12 = Value from SASSXXLX or zeros

Note: If this exit is invoked from a system task, there will be no saved value from SASSXXLX. The value of the word at +12 will be zero. Use EXITPARM EXIT=SASSXX04 to map exit parameters.

Output: R15 = 0 Issue the ENQ/RESERVE (QNAME may be changed)
 R15 Not equal to 0 Do not issue the ENQ/RESERVE

CA-7 uses a QNAME of SYSDSN.

9.3.5 Queue Entry JCL

Application Name: SASSXX05 SAMPJCL: UL23321

Exit Purpose: Used to add, delete, or change JCL statements or control statements at the time execution JCL is attached to a job in the Request queue. Also, the JQUSER field in the JQREC header (a 20-byte user-defined field) may be modified if desired. Any changes made to the JCL in this exit are reflected in the trailer queue JCL.

Caller: This exit is SLINKed from SASSSCJL and SASSSM50. This exit may be /RELINKed.

Module Attributes: This exit must use reentrant coding.

Input: R2 = Address of parameter list

- +0 = XL1 function code
 - X'00' - JCL record is being passed
 - X'04' - Return from add
 - X'08' - End of data
 - X'0C' - Wrap-up call
- +1 = AL3 address of 80-byte statement
- +4 = AL4 address of JQREC header
- +8 = AL4 4-byte exit work area (zero upon initial call for each job)
- +12 = Value from SASSXXLX or zeros

Note: If this exit is invoked from a system task, there will be no saved value from SASSXXLX. The value of the word at +12 will be zero. Use EXITPARM EXIT=SASSXX05 to map exit parameters.

Output: R15 =Return code

- 0 = Write this statement and keep processing (this statement may have been modified). If end of data is reached, then a 0 return tells CA-7 to close off the JCL stream and not add to it.
- 1 = Delete this statement from the JCL.
- 2 = Write the statement and return to the exit (used to add statements).
- 16 = Cancel the job.

The job name must not be changed. Addresses in the parameter list must not be changed.

Note: If hex data is put in the JCL, it will probably cause problems when displaying from the queues (for example, QJCL and LQ,LIST=JCL).

A job cannot be canceled in the wrap-up call. That is, a return code of 16 is not honored when the entry function code is X'0C'.

If a statement is bypassed due to #JI or #JO scheduled override parameters, the statement will not be passed to this exit.

9.3.6 External DSN Access

Application Name: SASSXX07 SAMPJCL: UL23323

Exit Purpose: For user examination and possible checking of external data set security. The exit is invoked when accessing a data set through CA-7 commands (for example, utilities, DB.7 screen).

Caller: This exit is called by SASSSSEC. This exit may be /RELINKed.

Module Attributes: This exit must use reentrant coding.

Input: R2 = Address of parameter list with the following values:

+0 = UID of terminal user

+1 = Access code:

 A = create (add/allocate)

 R = read

 S = scratch (delete)

 W = write

+2 = Return code from CA-7 security check, halfword of 0 if CA-7 allows access or halfword of 4 if CA-7 restricts access. (Note: If a USERID module is not being used, a return code of 0 (zero) is passed.)

+4 = Address of 44-byte dsname

+8 = Address of 8-byte member name or zero

+12 = Address of 6-byte volume serial

+16 = Value from SASSXXLX or zeros

+20 = 80-byte message area if RC not equal to 0.

Note: If this exit is invoked from a system task, there will be no saved value from SASSXXLX. The value of the word at +16 will be zero. Use EXITPARM EXIT=SASSXX07 to map exit parameters.

Output: R15 = 0 if access is allowed, or 4 if access is restricted. This return code must also be placed in the parameter list at offset+2 (see Input).

Note: When writing to an FWLP data set as a result of the FWLP command, and when logging to the browse data set, this exit is not invoked. When these data sets are accessed by data set name (for example; LPDS, JCL(FETCH) commands), they pass through this exit. Also, this exit should only be run under user or terminal SCTs.

This exit is invoked for the following:

- Commands:
DEMAND(H)
RUN(H)
LOAD(H)
LJCK
LJCL
LLIB
LPDS
- Any function of the DB.7 screen except CLEAR, DBM, EDIT.
- Utility commands that reference data set names:
AL/ALC
CAT
DMPCAT
DMPDSCB
DMPDSN
FIND
LISTDIR
LOC
MAP
RENAME
SCRATCH(P)
UNC

9.3.7 Forecast Worksheet

Application Name: SASSXX08 SAMPJCL: UL23324

Exit Purpose: Used to add text lines to Forecast Worksheets in the Report Summary portion.

Caller: SASSXX08 receives control from module SASSSFCW. This exit may be /RELINKed.

Module Attributes: This exit must use reentrant coding.

Input: R2 = Address of parameter list

- +0 = Address of job name
- +4 = Address of 74-byte output area
- +8 = JQREC base address if job in queue (queue record for job)
- +12 = SASJE base address (job data set member record)
- +16 = SASSJOBR base address (job data set directory record)
- +20 = Value from SASSXXLX or zeros

Note: Use EXITPARM EXIT=SASSXX08 to map exit parameters.

Output: R15 = Return code

- 0 = Do not write output area but continue processing
- 1 = Write output area and continue processing
- 2 = Write output area and return to exit (used to add lines)

9.3.8 Command Exit

Application Name: SASSXX09 SAMPJCL: UL23325

Exit Purpose: To allow user modification of CA-7 command input. The exit may support the use of command aliases.

Caller: SASSXX09 is called by SASSSSEC. This exit may be /RELINKed.

Module Attributes: The exit should use reentrant coding.

Input: R2 = Address of parameter list

- +0 = Value from SASSXXLX or zeros
- +4 = Halfword containing length of input
- +6 = 200-byte area containing input
- +206 = 80-byte area for message
- +286 = Halfword containing length of command output
- +288 = 200-byte area containing command output

Output: R15 = Return code

- 0 = Accept command (use input as received)
- 4 = Change command (use output area and length)

R15 not 0 or 4 = Reject command and issue message

Note: Use EXITPARM EXIT=SASSXX09 to map exit parameters.

9.3.9 Job Data Verification

Application Name: SASSXX10 SAMPJCL: UL23326

Exit Purpose:

To validate and or modify job data entered on the DB.1 - Job Definition panel for ADD, UPDATE, and DELETE functions and through the load process for jobs being added or replaced on the CA-7 database.

Caller: This exit is SLINKed (invoked) from modules SASSSM20 and SASSSJL1. This exit may be /RELINKed.

Module Attributes:

The exit must use reentrant coding.

Input:

R2 = Address of parameter list

+0 = Address of job directory record image (mapped by SASSJOBR)

+4 = Address of job member record image (mapped by SASJE)

+8 = Address of 8-byte message area.

On entry to exit this field contains one of the following strings to indicate the function and caller:

CL8'JOBSADD' - Add function on the DB.1 panel

CL8'JOBSREPL' - Update function on the DB.1 panel

CL8'JOBSDEL' - Delete or DD function on the DB.1 panel

CL8'LOADADD' - Add processing from job load

CL8'LOADREPL' - Update processing from job load

+12 = Value from SASSXXLX or zero.

Output:

R15 = return code

0 = Continues with the requested function without changes.

4 = When received from the DB.1 panel, the ADD/UPDATE/DELETE fails with a SM20-05 message and the 8-byte message field is displayed with the note that the user exit caused the failure. **This return code is ignored when the XX10 exit is entered during the job load process.**

8 = The user exit has modified data in the job directory and or the job member. The changes to the records are applied to the CA-7 database for the job entry. **It is the responsibility of the user exit to ensure that the modified data is in the correct format to prevent corrupted job records on the CA-7 database.**

When a return code of 4 is returned by the user exit from the DB.1 panel, the user receives the following message:

SM20-05 INVALID OR OMITTED xxxxxxxx DATA, REJECTED BY USER EXIT

The xxxxxxxx would be the 8-byte information supplied by the user exit.

This exit is invoked after CA-7 validates the UID access for the job. This exit is invoked by and for the following:

- The DB.1 screen ADD, UPDATE, DELETE, or DD function
- Job adds during the CA-7 job load process
- Job updates during the CA-7 job load process

Note: Use EXITPARM EXIT=SASSXX10 to map the input parameter list.

9.3.10 JCL Attach Verification

Application Name: SASSXX11 SAMPJCL: UL23327

Exit Purpose: Use to verify that JCL is being attached from the correct library. The job can be set to be left in SKELETON status or RETRY status in the Request queue, or standard JCL attach will continue.

Caller: This exit is SLINKed (invoked) from SASSSCSR. This exit may be /RELINKed.

Module Attributes: This module must use reentrant coding.

Input: R2 = Address of parameter list

+0 = Address of job queue record (mapped by macro JQREC)

+4 = Address of JCL table entry (mapped by macro JCLDSENT)

+8 = Address of job directory (mapped by macro SASSJOBR) or zeros

+12 = Address of job member (mapped by macro SASJE) or zeros

Note: If a job is entering the Request queue, but is not defined to the database, the addresses for job directory and job member will be zeros.

Output: R15 = 0 Continue with JCL attach

R15 =4 Do not attach JCL, but set the job to RETRY status (if RETRY is not on, the status will show as SKELETON).

R15 = Any other value prevents the JCL attach (the job displays SKELETON status in the request queue).

Note: Use EXITPARM EXIT=SASSXX11 to map the input parameters.

9.3.11 Database Load Processing

Application Name: SASSXX12

This exit is described in "Standard Exit Descriptions" on 9.4.2, "Database Load Processing" on page 9-35.

9.3.12 SMF Feedback

Application Name: SASSXX13 SAMPJCL: UL23329

Exit Purpose: SMF extracts retrieved from the CA-7 communications data set may be examined, modified, or deleted using the SASSXX13 exit.

Caller: SASSXX13 is entered through SLINK from SASSSMF0. SASSXX13 must return control to CA-7 through SEXIT. This exit may be /RELINKed.

Module Attributes: Calls to the exit are single threaded and occur under the control of the SASSSMF0 SCT. The exit should be reusable.

Note: Because SMF feedback processing is a function critical to the proper execution of CA-7, it is **STRONGLY** recommended that processing in SASSXX13 be limited to routines of short duration. Any activity which might result in a suspension or even slight degradation of SMF feedback processing will likely affect CA-7 performance adversely.

Input: R2 = Address of parameter list (mapped by EXITPARM)

 +0 = Return code value set by SASSXX13

 0 - continue processing with this SMF extract

 4 - allow this SMF extract to be modified

 8 - do not continue processing with this SMF extract

 +4 = Address of the SMF extract (see note below)

 +8 = Reserved for client use

 +12 = Reserved

 R10 = Address of SASSXX13 entry point. (Set by SLINK macro processing)

 R11 = Address of the SASSSMF0 SCT

 R12 = Address of the UCC7SVT

Note: R11 and R12 must not be modified by the SASSXX13 exit.

 R15 = Address of SASSXX13 entry point

Output: The return code in the parameter list pointed to at entry is used to determine the action to be taken on the SMF extract. The value set in R15 by SASSXX13 is ignored.

Related Messages:

If the SASSXX13 is used, the following WTO may be issued:

CA-7.SMF0 - BAD LRECL FROM SASSXX13, EXIT DISABLED

This indicates that the LRECL of a record modified by SASSXX13 was less than 8 or greater than 1024. If this condition occurs, no further calls are made to the SASSXX13 exit.

Additional Notes:

For information on the format of the SMF extracts, see the SASS7LOG macro, record types 04, 05, 0E, 0F, 14, 76, and 90-9F.

9.3.13 Personal Scheduling Verification

Application Name: SASSXX14 SAMPJCL: UL23330

Exit Purpose: Function requests entered on the CA-7 Personal Scheduling screen can be examined and accepted or rejected by this exit. If rejected, the exit can pass a 60-byte message to be displayed on the user screen. This allows you to enforce local standards in the Personal Scheduling facility. For example, you may want all Personal Scheduling jobs to have a T in the fourth position of the job name. Or, you may want to restrict which JCL-IDs can be accessed from Personal Scheduling.

Caller: Called from SASSPS00. This exit may be /RELINKed.

Module Attributes: This exit must be coded reentrant.

Input: R2 = Address of parameter list

- +00 = Value from SASSXXLX logon exit or zero.
- +04 = Address of 8-byte field containing screen function (ADD, DELETE, LIST, STATUS, SUBMIT, or UPDATE)
- +08 = Zero or address of 8-byte field containing job name
- +12 = Zero or address of 2-byte field containing JCL-ID (in hex)
- +16 = Zero or address of 8-byte field containing pattern job name
- +20 = Zero or address of 2-byte field containing time (in hex, hhmm)
- +24 = Zero or address of 8-byte field containing predecessor
- +28 = Address of 60-byte field for error message if the exit rejects the request
- +32 = Zero or address of 16-byte field containing JCLLIB

Output: R15 = Return code

0 = Allow processing for the request to continue.

Not 0 = Reject the request. The user receives the following messages:

```
'PS00-16 REQUEST REJECTED BY SASSXX14 USER EXIT'  
'... xx14 error message (if set by exit).....'
```

Note: This exit can only accept or reject personal scheduling requests, the request itself *cannot* be changed by the exit.

The exit is invoked after the FUNCTION has been validated, and a security check has been made to ensure the current user has authority to issue that function on the Personal Scheduling screen. Validation of any other fields will be performed only after the call to SASSXX14 has been made. The request may eventually be rejected even though the user exit has accepted it.

Note: Use EXITPARM EXIT=SASSXX14 to map the input parameters.

9.3.14 CA-7 Console Terminal Output

Application Name: SASSXX15 SAMPJCL: UL23338

Exit Purpose: Use this exit to monitor the output to the CA-7 console terminal. It can allow the original WTO to be issued or suppressed.

Caller: Called by SASSWCSL. This exit cannot be /RELINKed.

Module Attributes: This exit must be coded reusable but not necessarily reentrant. However, any waits issued, explicit or implicit, cause the entire CA-7 system to wait.

Input: R2 = Address of parameter list
 +0 = Address of WTO list in the form MF=(E,(1))

Output: R15 = Return code
 0 = Do the original WTO
 Not 0 = Suppress the WTO

The APPLCTN statement for SASSXX15 must use ATTR=PERM or ATTR=RESD.

9.3.15 SMF WTO Message

Application Name: SASSXX16 SAMPJCL: UL23339

Exit Purpose: Use this exit to monitor CA-7.SMF3 and/or CA-7.SMF4 WTOs that are generated. These optional WTOs are generated to indicate unsuccessful job/step terminations. The exit can allow the original WTO to be issued or suppress it.

Caller: Called by SASSSMF3 and SASSSMF4 using BALR R14,R15 (NOT SLINK). Registers 0 - 14 are saved prior to calling the exit and restored on return. This exit cannot be /RELINKed. The exit is called by SASSSMF3 for WTO CA-7.SMF3 if the WTO= option is set on the CA-7 RESTART initialization statement. The exit is called by SASSSMF4 for WTO CA-7.SMF4 if the WTOSTEP= option is set on the CA-7 RESTART initialization statement.

Module Attributes: This exit must be coded reusable, but not necessarily reentrant. However, any waits issued, explicit or implicit, will cause the entire CA-7 address space to wait.

The APPLCTN statement for SASSXX16 must use ATTR=PERM or ATTR=RESD. Code in this user exit should be kept to a minimum to ensure that the timing of the SMF feedback process in CA-7 is not degraded.

Input: R2 = Address of parameter list

+0 =	Address of a copy of the SMF extract (mapped by macro SCOMREC)
+ 4 =	Address of a copy of the job queue record (mapped by macro JQREC)
+ 8 =	Address of a copy of the WTO in MF=(E,(1)) format
+12 =	Reserved for client use
+16 =	Reserved for future use

Output: R15 = Return code

0 =	Issue the WTO
Not 0 =	Suppress the WTO

9.3.16 Browse Message

Application Name: SASSXX17 SAMPJCL: UL23342

Exit Purpose: Use this exit to monitor messages written to the CA-7 Browse data set. The exit cannot change or suppress Browse messages; however, it may issue its own WTOs.

Caller: Called by SASSDLOG using SLINK. The exit must return control through SEXIT.

Module Attributes: This exit must be coded reusable, but not necessarily reentrant. However, any waits issued, explicit or implicit, may impact the performance of the CA-7 system.

The APPLCTN statement for SASSXX17 should use ATTR=PERM or ATTR=RESO, but this is not required.

Input: R2 = Address of parameter list
 +0 = Address of 80 byte Browse message
 + 4 = Reserved for client use
 + 8 = Reserved for future use

Output: None

Note: Use EXITPARM EXIT=SASSXX17 to map the input parameters.

9.3.17 Logon/Password Verification

Application Name: SASSXXLX SAMPJCL: UL23313

Exit Purpose: May be used to monitor logons at CA-7 terminals. This exit is invoked prior to CA-7 or external security checking. If the exit accepts the logon, then CA-7 or external security may yet reject the logon. If the exit rejects the logon, then additional security validation is not performed.

Caller: SASSXXLX is called by SASSSSEC. This exit may be /RELINKed.

Module Attributes: The exit must be coded reentrant. However, any waits issued, explicit or implicit, cause the entire CA-7 system to wait.

Input: R2 = Address of parameter list

- +0 = Value that may be passed to other exits
- +4 = 8-byte USERID
- +12 = 8-byte password
- +20 = 8-byte new password
- +28 = 8-byte CA-7 terminal name
- +36 = 8-byte LU name (VTAMID)
- +44 = 8-byte APPL value from SECURITY statement
- +52 = 40-byte PARMS from LOGON screen or command
- +92 = 80-byte message area for logon errors

Note: Use EXITPARM EXIT=SASSXXLX to map exit parameters.

Output: R15 = Return code

- 0 = Accept the logon
- 2 = Accept the logon and allow the userid, password, and new password fields to be changed
- 6 = Accept the logon and skip CA-7's external security check
If this return code is to be used, the EXTERNAL keyword should be omitted or only specify LOGON
- Not 0 or 2 or 6 = Reject the logon and display message

If the exit rejects the logon, an 80-byte message will be displayed. The text of this message is provided by the exit by modifying an area at R2 +92.

Even if the SASSXXLX exit is used to modify the userid of the person logging on to CA-7, the original userid is still used in the history reports produced by program SASSHIS8.

9.3.18 Logoff Exit

Application Name: SASSXXFF SAMPJCL: UL23314

Exit Purpose: To process /LOGOFF from CA-7 terminals. The exit receives control when a /LOGOFF command is entered. SASSXXFF does not receive control when the VLOGOFF PF key is pressed.

Caller: This exit receives control from SASSSSEC. This exit may be /RELINKed.

Module Attributes: This exit must use reentrant coding.

Input: R2 = Address of parameter list

- +0 = Value from SASSXXLX or zeros
- +4 = 8-byte USERID
- +12 = 8-byte CA-7 terminal name
- +20 = 8-byte LU name (VTAMID)

Output: R15 is ignored

Note: Use EXITPARM EXIT=SASSXXFF to map exit parameters.

Since SASSXXFF may not be called for every condition that can cause a terminal to be logged off (or disconnected), it is called at logon time before the SASSXXLX exit is called. This is to allow for any possible cleanup that may be required. This means that this exit is usually called twice for each logged on session: once when a "normal" logoff occurs and again when the next session is about to occur.

9.3.19 Dump Queue

Application Name: SASSXXQD (DMPQ) SAMPJCL: UL23315

Exit Purpose: To allow addition, deletion, modification, or capture of CA-7 queue records before records are unloaded to the CA-7 dump queue file.

Caller: SASSXXQD is SLINKed from SASSINQD. The exit may be /RELINKed but is only processed once at /SHUTDOWN.

Module Attributes: This exit must be serially reusable.

Input: R2 = Parameter list

+0 = Address of control fields

+0 = Return code (2 bytes)

H'0' = Normal processing

H'4' = Delete current record

H'8' = Do not return

H'12' = Insert record pointed to by record address

+2 = Communication code (2 bytes)

H'0' = Subsequent calls

H'4' = First call

H'8' = End of data

+4 = Date of DMPQ operation (x'0cyydddF')

+8 = Time of DMPQ operation (x'hhmmssth')

+12 = File status

X'80' DMPQ on disk

X'01' DMPQ not complete. (Check only when communication code is H'8'.)

+4 = Record address of JQREC

Output: Input parameter list with return code filled in and record address changed for inserts.

Note: Use EXITPARM EXIT=SASSXXQD to map exit parameters.

9.3.20 Move Queue

Application Name: SASSXXQB (MOVQ) SAMPJCL: UL23341

Exit Purpose: To allow addition, deletion, modification, or capture of CA-7 queue records before records are reloaded to the CA-7 queues.

Caller: Called from SASSINQB. May not be relinked.

Module Attributes: This exit must be serially reusable. No APPLCTN statement is needed for this exit. Control should be returned with a BR R14 and not an SEXIT.

Input: R13 = Address of 18 word save area
 R14 = Return address
 R15 = Address of SASSXXQB entry point

Output: Input parameter list with return code filled in and record address changed for inserts.

9.4 Standard Exit Descriptions

9.4.1 Batch Card Load Program Exit

Application Name: SASSBCX1

The Batch Card Load Program (BCLP) provides a means of creating card-image data sets which may be input requirements for CA-7 jobs. Through BCLP, data sets may be created (CREATE), records in a data set may be replaced (REPLACE) or records may be added (MODDATA).

Exit Purpose: To receive control after each record is read to perform one of the following:

- Continue normally
- Delete a record
- Replace a record
- Insert a record
- Bypass the data set request

Input: R1 = Address of a parameter list with the following format:

+0 = 4-byte address of the card image
 +4 = 4 bytes of zeros
 +8 = 4 bytes of zeros
 +12 = 4 bytes of zeros

The exit is responsible for saving and restoring all registers.

Output: On return from the exit, the parameter list may contain the following:

+0 4-byte address of card image which must be left as is at entry
 +4 Address of a card image to replace the record passed or to be inserted in front of the record passed
 +8 4-byte return code. The first 3 bytes should be zeros. The last byte must contain one of the following return codes:
 X'00' - normal completion
 X'04' - delete the record passed
 X'08' - replace the record passed
 X'0C' - insert a record
 X'10' - bypass the remainder of the data set request
 +12 Address of a line to print. This additional message must be 80 characters in length.

The action of this exit may be taken for any record including control statements. When control statements are modified, take special care to avoid errors. The routine may also request that an additional message be printed with the normal listing.

9.4.2 Database Load Processing

Application Name: SASSXX12

Exit Purpose: Records used to update the CA-7 database created by the SASSJJCL program may be examined, modified, or deleted by this exit. One possible use of the exit would be to examine a DSN to be loaded in the CA-7 database. If the DSN to be loaded matched a DSN in a site defined table, then the exit could force the data set to be identified as PERM in CA-7.

Caller: Called from SASSJJCL. May *not* be RELINKed.

Module Attributes: This module need not be reentrant. It is LOADED once per execution of SASSJJCL. This exit observes standard MVS linkage conventions.

Input: R1 = Address of parameter list (mapped by EXITPARM)

+0 = Return code value set by SASSXX12

0 - write the load record

4 - do not write the load record

+4 = Value of R1 at entry to SASSJJCL

+8 = Address of the ICMDSCT used by SASSJJCL

+12 = Code indicating the type of IBM record currently examined by SASSJJCL:

1 - JCT

2 - SCT

3 - SIOT/JFCB

4 - SSWA

+16 = Address of the IBM record currently examined by SASSJJCL

+20 = Type of CA-7 database load record

+24 = Address of CA-7 database load record

If this value is zero, then SASSXX12 is called to allow examination of the IBM records used by SASSJJCL, and the return code set by SASSXX12 for these calls is ignored.

If this value is not zero, then this word points to the CA-7 database load record which can be modified. The SASS7LOG macro may be used to map CA-7 database load records.

Note: The length of the record must not be modified.

+28 = One word reserved for client use

R13 = Address of standard 18 fullword save area

R14 = Return address

R15 = Address of SASSXX12 entry point

Output: The return code in the parameter list pointed to at entry is used to determine whether the CA-7 database record is written. The value set in R15 by SASSXX12 is ignored.

Related Messages:

If SASSXX12 is used, the following summary messages are produced at SASSJJCL termination:

```
CA-7.112  xxxxxxxx  LOAD RECORDS CHANGED BY SASSXX12
CA-7.113  yyyyyyyy  LOAD RECORDS DELETED BY SASSXX12
```

where xxxxxxxx is the total number of CA-7 database load records modified and yyyyyyyy is the total number of CA-7 database load records deleted by the SASSXX12 exit.

Additional Notes:

For information on the format of the records created by the SASSJJCL program, see the SASS7LOG macro, record types 90, 91, 92, 93, and 94.

9.4.3 Problem Management Interface

Application Name: Client specified. Declared as the EXIT keyword value on the NETMAN statement in the CA-7 initialization file.

Exit Purpose: Allow inspection of CA-7 job completion data for problem tracking.

If CA-Netman Version 4.9 or greater is used, then this exit is not needed and should not be specified. This is because there is an internal exit used and specifying it here would disable the automatic existing exit. The NETMAN statement in the initialization file is still needed if the interface is desired, but the exit should not be specified. If a version prior to Version 4.9 of CA-Netman is used or some other problem management interface is used, then this exit may be used to get control and inspect CA-7 job completion data.

Caller: Called from SASSPM00. May not be /RELINKed. The module is LOAded at CA-7 initialization and control is passed using BASR R14,R15. Because the module is called from the CA-7 Problem Management Subtask, CA-7 Main Task services such as SGETM and SWAIT will not be available. Use OS/390 services instead.

Module Attributes: The module need not use reentrant coding. It should be linked AMODE 31, RMODE ANY.

Input: R1 = Address of area mapped by the DSECT L2PM.
 This DSECT is generated by the L2PMA macro.
 The area contains information about the
 completion of a CA-7 submitted job.

 R12 = Address of UCC7SVT

 R13 = Address of 18 word savearea

 R14 = Return address

 R15 = Module entry point address

Output: There is no significant output. The return code in R15 is ignored.

Note: Job completion information is passed to the exit in the order that jobs complete.

If CA-Netman is used for problem management, this exit point is not available.

If GOODCOMP=N is coded on the NETMAN statement in the CA-7 initialization file, only information on abnormal job completions and their normally complete restarts is passed to the exit (this is the default). If GOODCOMP=Y is coded, information on all CA-7 job completions is passed to the exit.

9.4.4 Database Backup

During database backup processing, a user exit may be used to examine the records being written to the backup tape. No changes should be made to the records; however, writing copies of the records to some user-defined data set for subsequent analysis, customized reporting, and so forth, may be of value to the user.

Module SASSBK00, which performs the backup process, uses PARM data to identify the functions which are to be performed. Keyword EXITMOD may be coded in the PARM to identify the user exit which is to be used whenever an OPT=BACKUP is being performed. The exit may reside in a JOBLIB, STEPLIB or link list library. See the "Backup and Recovery Considerations" chapter of this manual for an example of the JCL required for this program. (See 7.1.1.4, "SASSBK00" on page 7-6 and 7.1.1.5, "SASSBK00 DD Statements" on page 7-7.)

Input: A parameter list address is passed to the exit program in register 1. This address points to a list of fullword addresses as follows:

+0 A(function)

1-byte character field

B = Backup being performed, record being passed
 R = Reload being performed
 T = Tapeonly run for reporting
 E = End-of-data call

+4 A(dstype)

3-byte character field

JOB = Record from job data set
 DS = Record from dataset data set
 IDS = Record from index data set

+8 A(rectype)

3-byte character field

DIR = Record is a directory type of record
 MEM = Record is a member type of record
 IDS = Record is an index type of record

+12 A(record)

4-byte address of the record

Binary zeros when function E (end-of-data) is passed.

+16 A(keyad)

4-byte address of current element key

Each record coming through the exit is a logical record in the database and looks the same regardless of the form of the database.

Records are mapped as follows:

- For rectype=IDS records are mapped by SASSCTG macro.
- For rectype=DIR records are mapped by SASSJOBR macro.
- For rectype=MEM the following applies:
 - For dstype=JOB use macro SASJE
 - For dstype=DS mapping by:
 - DSRECD if DS member
 - SASSNTWK if NW member
 - SASSPROS if PP member
 - SASSIJO if SI, SJ, SO member

9.4.5 Database Reload

During database reload processing, a user exit may be used to examine the records being read from the backup tape. This exit can also be utilized for reporting-type of runs which use only the backup tape without reloading the database. The exit capabilities are essentially the same as for a backup exit as discussed earlier.

The exit is governed by the same rules as the exit for 9.4.4, “Database Backup” on page 9-38. The same PARM keyword, EXITMOD, is used to identify this exit. It is given control whenever OPT=RELOAD or DB=TAPEONLY functions are performed by SASSBK00.

9.4.6 Database Cross-Reference

Two cross-reference reports are provided by the system:

SASSXREF-01 on 9.4.6.1, “Program-Job Cross Reference” on page 9-41

SASSXREF-02 on 9.4.6.2, “DSNAME-Program Cross Reference” on page 9-42

These are obtained by specifying OPT=XREF on the EXEC statement PARM for SASSBK00, the backup/reload program.

If a user-coded module is to receive control for cross-reference reporting, the XREFMOD default must be overridden with the name of the user module. See 7.1.1.6, “SASSBK00 PARM Values” on page 7-9 and 7.1.1.4, “SASSBK00” on page 7-6.

9.4.6.1 Program-Job Cross Reference

This report provides the name of every program executed in CA-7 controlled jobs and the name of every job in the CA-7 database which includes the program. It can be very helpful whenever a program is to be changed or deleted by ensuring that all references to the program can be considered.

SASSXREF-01		C A - 7 - PROGRAM-JOB CROSS REFERENCE					PAGE 001	
RUN DATE 05/31/yy								
PGM NAME	JOB NAME	---STEP--- NAME NO.	--PROC--- STEP NAME	NO. RUNS	--LAST DATE	RUN-- TIME	SYSTEM NAME	
CC	C495MM1S	S1 001		4	01/27/yy	1613		
		S2 002		4	01/27/yy	1613		
		S3 003		4	01/27/yy	1613		
		S4 004		4	01/27/yy	1613		
		JOBCHECK 005		4	01/27/yy	1613		
EARL	D463DPCH	EARL 005		5	04/07/yy	1251	DISPATCH	
IEBGENER	BATCH001	JSTEP2 002		4	04/06/yy	1624	CA-7	
	CA07#SCC	STEP1 001		1	02/29/yy	1423	CA-7	
	CA07LOGP	SUBMIT 001		150	05/31/yy	0859	CA-7	
	CA07LOGS	SUBMIT 001		145	05/27/yy	1616	CA-7	
	D463DPCH	GENER 004		5	04/07/yy	1251	DISPATCH	
	D463ICOM	SUBMIT 001		1	02/13/yy	1339		
	D463XX01	GEN1 001	S1	4	04/01/yy	1204	TESTNTWK	
		GEN2 003	S1	4	04/01/yy	1204	TESTNTWK	
	D463XX02	GEN1 001	S1	4	04/01/yy	1209	TESTNTWK	
	D463XX03	GEN1 001	S1	4	04/01/yy	1209	TESTNTWK	
	D463XX04	GEN1 001	S1	2	04/01/yy	1209	TESTNTWK	
	D463XX05	GEN1 001	S1	3	04/01/yy	1210	TESTNTWK	
		GEN2 002	S1	3	04/01/yy	1210	TESTNTWK	
	D463XX06	GEN1 001	S1	2	04/01/yy	1211	TESTNTWK	
	D463XX07	GEN1 001	S1	1	02/03/yy	1422	TESTNTWK	
		GEN2 002	S1	1	02/03/yy	1422	TESTNTWK	
	D463XX08	GEN1 001	S1	3	05/28/yy	1041	TESTNTWK	
		GEN2 003	S1	3	05/28/yy	1041	TESTNTWK	
	D463XX09	GEN1 001	S1	2	02/03/yy	1418	TESTNTWK	
	D463XX10	GEN1 001	S1	2	02/03/yy	1419	TESTNTWK	
		GEN2 002	S1	2	02/03/yy	1419	TESTNTWK	
	GENR2CMS	PRINT 001		1	05/20/yy	1214	JACK	
	GENR2PRT	STEP1 001		3	03/10/yy	0957	GENER	
	JLTEST	PRINT 001		5	04/07/yy	1508		
	PURGERPT	SUBMIT 001		31	05/16/yy	2100	RPT-QA	
	U7TESTSV	D463SVCT	STEP1 001					

Figure 9-1. Program-Job Cross Reference Report

9.4.6.2 DSNNAME-Program Cross Reference

This report provides the name of each data set defined in the CA-7 database and itemizes the job steps which reference the data sets. It provides a quick reference for determining every JCL reference to a production data set.

SASSXREF-02		C A - 7 - DSNNAME-PROGRAM CROSS REFERENCE										PAGE 001	
RUN DATE 05/31/yy													
DSNAME	DSNBR	ORG	RFM	LRECL	BLKSI	DEVC	GDG	PGM NAME	JOB NAME	---STEP--- NAME NO.	----DD----- NAME NO.	DISP	
A.B	000042	??		00000	00000	DASD	NO						
A.B.C	000043			00000	00000	DASD	NO						
A.BB	000041			00000	00000	DASD	NO						
D402.EDIT.INPUT	000103			00000	00000	DASD	NO						
D463BD1.LOADLIB2	000018	PO	U	00000	15476	DASD	NO	SASSKFWX	CA7FLOW1	JS10	001 STEPLIB	001	SHR
D496MS1.JCL.TEMP2	000104			00000	00000	DASD	NO						
JACJA02.JCL.LIB1	000108			00000	00000	DASD	NO	IEBGENER	GENR2CMS	PRINT	001 SYSUT1	003	SHR
JACJA02.PDS.B133	000010	PS	FBA	00133	32718	DASD	NO	SASSHIS8	CA7HIS50	HLOGS	001 SASSRA01	018	SHR
								SASSHIS8	CA7HIS50	HLOGS	001 SASSRA02	019	SHR
								SASSHIS8	CA7HIS50	HLOGS	001 SASSRA03	020	SHR
JACJA02.PDS.MASK133	000034	PS	FBA	00133	32718	DASD	NO	IEBGENER	BATCH001	JSTEP2	002 SYSUT2	003	SHR
JACK1	000098			00000	00000	DASD	NO						
QA.PROD.LINKLIB	000100	PO	U	00000	06144	DASD	NO	EARL	D463DPCH	EARL	005 STEPLIB	001	SHR
SSDDEV.CA07.R32.BOUTPUT	000001	PS	FBA	00133	32718	DASD	NO	SASSHIS8	CA7BROWS	HLOGS	001 SYSLIST	013	SHR
SSDDEV.CA07.R32.BTEST	000072	PS	FB	00080	00080	DASD	NO	IEBGENER	GENR2PRT	STEP1	001 SYSUT1	002	SHR

Figure 9-2. DSNNAME-Program Cross Reference Report

9.5 Changing Graph Definitions

The APA facility provides numerous graphs on CA-7 performance and activities. Users can also define their own graphs, modify graph titles, and so forth, that are provided with the system. All definitions/redefinitions must be made in the APA graph tables.

9.5.1 Graph Table Definition

Graphs to be referenced through APA are predefined and link edited to the CA-7 Load library. The object module from this assembly has been link edited with the following standard name (one for each of the categories of graphs):

SASSGPHx

Where:

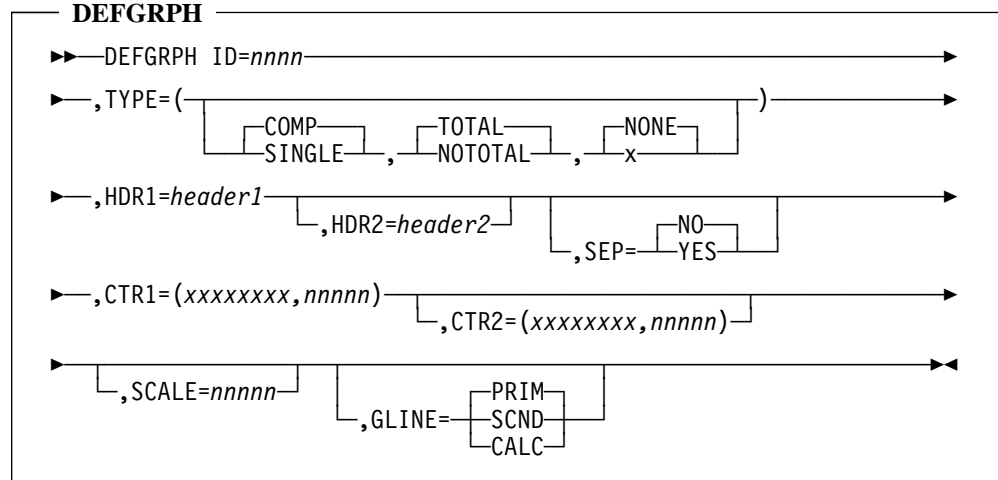
x = D - Database graphs table
J - Job graphs table
N - Network graphs table
S - System graphs table

9.5.2 DEFGRPH Macro

Use the DEFGRPH macro in the SASSGPHx modules to define graph tables used in the Automated Performance Analysis (APA) application.

9.5.2.1 Syntax

Use the following format to generate each graph definition:



Where:

ID

Indicates a required identification number, up to 4 digits, assigned to each graph. Must be unique only within each graph definition table. When coding multiple graph definitions, this identification number must be in ascending numerical sequence in regard to other DEFGRPH macros.

TYPE

An optional parameter defining how the statistics are to be reported. Consists of three subparameters enclosed in parentheses.

Indicates type of graph as follows:

COMP

Comparison graph (2 counters required). This is the default.

SINGLE

Single counter graph.

Indicates whether a total line should appear as follows:

TOTAL

Formatted total line. This is the default.

NOTOTAL

No formatted total line.

Indicates type of calculation as follows:

NONE

No calculation. This is the default.

x

Values are:

A

Addition (CTR1+CTR2)

D

Division (CTR1/CTR2)

R

Running Total (for SINGLE graphs only)

P

Percentage (CTR2/CTR1)

S

Subtraction (CTR1-CTR2)

N

Negative replacement (CTR2=CTR1-CTR2, percentage calculation assumed)

HDR1

Is a required heading, up to 35 characters, to appear as the title line of the graph. Must be enclosed in single quotes.

HDR2

Is an optional heading, up to 35 characters, to append to the value defined for HDR1 to make a title longer than 35 characters. Must be enclosed in single quotes.

SEP

Is an optional specification to request the 4 characters bVSb (versus) as a separator between the character values of HDR1 and HDR2. If NO separator is requested, the two heading values are concatenated, therefore allowing up to 70 positions of heading data. The default is NO.

CTR1

Is a required parameter describing the primary counter used in the graph. Consists of two subparameters enclosed in parentheses.

xxxxxxx

Is a required assembler label, up to 8 characters, describing a field name found in the STATAREA macro DSECT. The length attribute of the field name is calculated by the assembler and must be 4 bytes, 2 bytes, or 1 byte in length. Any other specification causes unpredictable results.

nnnnn

Is an optional numeric value in the range from 1 to 99999 describing the division factor for the graph. This could be used to convert seconds to minutes, minutes to hours, and so on, if the value accumulated in the counter is not what is desired for the graph. Default is 1.

CTR2

Is required only if TYPE=COMP. Describes secondary counter used in comparison graph. CTR2 is coded in same format as CTR1.

SCALE

Is an optional parameter used to supply the default increment value in calculating the length of the graph scale. Must be a numeric value in the range from 1 to 99999. The default is 2.

GLINE

Is an optional parameter describing the value to represent on the output graph line. Must be one of the following:

PRIM

Use primary counter for graph line. This is the default.

SCND

Use secondary counter for graph line.

CALC

Use calculated value for graph line.

9.6 Adding Counters for New Graphs

Counters have been provided to allow the user to accumulate up to 10 items of information besides those already being accumulated by CA-7. Reserved are 10 halfword user counters (STUSER1 through STUSER10) that are available to the user for creating new reports. Additional graphs for these counters may be added by supplying the required information in the appropriate graph table module.

9.6.1 JCL Attach Exit Example

The following is an example of a user exit (SASSXX05) that uses the BMPSTAT macro to increment a counter (STUSER1) in the STATAREA.

Suppose a graph is desired that displays the number of jobs scheduled from Department XX and the originating department can be determined by examining positions 3 and 4 of the job name. The following program could be used for the SASSXX05 JCL attach exit:

```

SASSXX05  START
          PRINT  NOGEN
          UCC7SVT
          SASSEQU
          STATAREA
          JQREC
SASSXX05  CSECT
          SASSVRSN REGS=R10          SET BASE ADDRESSABILITY
          L        R3,4(R2)          JQREC HDR
          USING    JQREC,R3
          CLI      8(R2),0            INITIAL CALL
          BNE      ENDXX05            B IF NOT
          MVI      8(R2),255          SET INDICATOR
          CLC      JQJNAME+2(2),=C'XX' DEPARTMENT XX
          BNE      ENDXX05            B IF NOT
          BMPSTAT  F=STUSER1,T=H,NOSTAT=ENDXX05  ADD 1 TO HALFWORD
                                          COUNTER 'STUSER1'
*
ENDXX05   SR      R15,R15            SET RETURN CODE
          SEXIT
          END

```

9.6.2 Defining Single Graphs

After assembling and link editing the exit, you should modify, assemble, and link edit SASSGPHJ after inserting a DEFGRPH macro to define the desired graph.

9.6.2.1 Examples

10	16	72
DEFGRPH ID=8743,		X
TYPE=(SINGLE,NOTOTAL,R),		X
SCALE=5,		X
CTR1=(STUSER1,1),		X
HDR1='JOBS SCHEDULED BY DEPT XYZ'		X

9.6.3 Defining Comparison Graphs

Instead of just reporting the number of jobs from Department XX, suppose you wanted to find out the percentage of total jobs scheduled versus those scheduled by Department XX only. This would involve a comparison graph. The procedures would be the same as those given for a single counter graph except the DEFGRPH macro would be similar to the following example.

9.6.3.1 Examples

10	16	72
DEFGRPH ID=7510,		X
TYPE=(COMP,TOTAL,P),		X
SCALE=2,		X
CTR1=(STSCHDS,1),		X
CTR2=(STUSER1,1),		X
GLINE=CALC,		X
HDR1='TOTAL JOBS SCHEDULED',		X
HDR2='JOBS SCHEDULED BY DEPT XYZ'		X
SEP=YES		X

9.7 Other System Modification Techniques

Besides user exits, there are alterations that may be made by a user. These are:

- Batch Interface Message Table modifications
- Generic Unit Name/Device Type Table modifications
- Message Level Suppression Table modifications
- Security modifications (See the *CA-7 Security Guide* for more information.)
- CA-Driver customization

9.7.1 Batch Interface Message Table

Application Name: SASSXXBT SAMPJCL: UL23312

The batch terminal interface (BTI) facility and the CA-7 CCI Batch Interface allow CA-7 commands to be issued from a batch job rather than an online terminal. See "External Communicators" in the *CA-7 Interfaces Guide* and "Using Batch Terminals" in the *CA-7 Database Maintenance Guide* for more information.

Exit Purpose: The SASSXXBT Message Table is used by the CA-7 Batch Terminal Interface process (BTI) and the CA-7 CCI Batch Interface to detect error and warning messages in batch output and to set a nonzero return code for the BTI or CCI Batch job step.

The SASSXXBT module is not executable. It is a table of messages (or message prefixes) and condition codes to be associated with those messages. When the batch terminal interface (BTI) program or CA-7 CCI Batch Interface program transfers the CA-7 command output from the output data set to SYSPRINT, each line is checked against the SASSXXBT table for matches. If an output line is matched with the table an information message is written to the ERRORS DD noting the text of the line that was matched and the associated condition code from the table.

At the end of BTI or CA-7 CCI Batch Interface processing, if any output lines were matched against the table a WTO is issued indicating the number of messages matched and the highest condition code. The return code for the batch step is then set to the highest condition code matched.

Coding:

The SASSXXBT table module is coded using the \$L2BTI macro which is in the CA-7 Macro library (CAIMAC). The assembled module must be in a load library accessible to the BTI program SASSBSTR and the CA-7 CCI Batch Interface program CAL2X2WB (that is, in the STEPLIB concatenation for the BTI and CCI batch steps). Each use of the macro defines a separate message in the table. An online example of the source for a SASSXXBT table is in the CA-7 SAMPJCL library provided on the CA-7 distribution tape. A coding sample also follows:

```

SASSXXBT START
*-----*
*   SAMPLE  SASSXXBT  ERROR  MESSAGE  TABLE                               *
*-----*
*   INDIVIDUAL MESSAGES ARE DEFINED USING THE $L2BTI MACRO.  EACH          *
*   ENTRY HAS THE MESSAGE TEXT (MSG=), AND THE CONDITION CODE TO BE        *
*   ASSOCIATED WITH THAT MESSAGE (CC=).                                     *
*-----*
*   THE MESSAGE TEXT (MSG=) IS REQUIRED.  THIS TEXT WILL BE COMPARED        *
*   WITH EACH LINE OF BATCH OUTPUT STARTING IN COL 1 FOR THE LENGTH        *
*   OF THE MSG= TEXT.  THE LENGTH OF THE MSG= TEXT CAN BE AS LONG OR      *
*   AS SHORT AS YOU WISH, BUT IT MUST EXACTLY MATCH THE BATCH OUTPUT      *
*   LINE IN ORDER TO BE CONSIDERED A HIT.  IF THE TEXT CONTAINS           *
*   IMBEDDED BLANKS OR COMMAS IT MUST BE ENCLOSED IN QUOTES (THE          *
*   QUOTES WILL NOT BE USED IN THE COMPARES).                             *
*-----*
*   THE CONDITION CODE (CC=) IS NOT REQUIRED.  IF OMITTED IT WILL           *
*   DEFAULT TO 4.  YOU CAN USE ANY NUMBER FROM 1 TO 255, EXCEPT 8,      *
*   16, AND 24 WHICH ARE RESERVED.                                         *
*-----*
*
*- THIS ENTRY IS FOR THE JOB NOT FOUND MESSAGE FROM AN LJOB COMMAND
*- (NOTE: SINCE THERE IS NO CC= THE CONDITION CODE DEFAULTS TO 4)
*
      $L2BTI  MSG=SLIA-02
*
*- THIS ENTRY IS FOR THE DSN NOT FOUND MESSAGE FROM AN LDSN COMMAND
*- (NOTE: THE CONDITION CODE IS EXPLICITLY SET TO 5.)
*
      $L2BTI  CC=5,MSG=SLIB-02
*
*- THIS ENTRY IS FOR THE MESSAGE YOU GET WHEN TRYING TO ADD A JOB
*- THAT ALREADY EXISTS.  (NOTE: SINCE THE TEXT HAS IMBEDDED BLANKS
*- IT HAS TO BE ENCLOSED IN QUOTE MARKS.)
*
      $L2BTI  CC=7,MSG='SM20-07 JOB ALREADY EXISTS'
*
*- THIS ENTRY IS FOR THE MESSAGE YOU GET WHEN TRYING TO DEMAND A
*- JOB THAT IS NOT DEFINED.  (NOTE: THERE ARE TWO
*- BLANKS BETWEEN THE MESSAGE ID AND THE MESSAGE TEXT.  IT HAS TO
*- BE DEFINED THIS WAY TO EXACTLY MATCH THE TEXT IN THE OUTPUT.)
*
      $L2BTI  CC=7,MSG='SP07-21 JOB NOT DEFINED AND JCLID'
*
      END

```

9.7.2 Generic Unit Name/Device Type Table

SASSUTBL provides a relationship between a generic unit name and an actual device type. A sample SASSUTBL is distributed with the system. It should be listed and modified as necessary to meet installation requirements for generic unit names. Each entry in the unit/device code table consists of a generic unit name in up to 8 characters followed by a 4-byte hexadecimal device code. For example:

DC	CL8'3330',X'30502009'
----	-----------------------

The device code (X'30502009' in the example) must match a device code in IBM's device name table.

SASSUTBL is used by CA-7 for the LOAD process, for analyze commands, and for utility commands such as CAT. Consequently, information about data sets in the database that is taken from this table is displayed for forecast and general inquiry commands and is used by workload balancing.

See 3.7.5, “Defining the Types of Tape Drives” on page 3-53 for a further discussion of SASSUTBL.

See UL23303 in SAMPJCL for example of SMP USERMOD.

9.7.3 Message Level Suppression

Due to the large number of messages generated by CA-7, a message matrix is provided which allows suppression of certain messages. This matrix is defined by SASSMSGs and may be changed to fit installation needs. However, only those messages defined in the distributed version of this module may be changed. No new messages may be added to the module. Also, messages are not routed based on these entries. Only suppression occurs.

The steps required for message suppression are:

1. Obtain a listing of source member SASSMSGs.
2. Change the SEND macros as desired.
3. Assemble and link edit SASSMSGs.
4. Shut CA-7 down and WARM start.

9.7.3.1 Examples

Completely suppress SP06-10 (station cancel message).

```
SEND M=SP0610B,LT=*NONE*
```

Only send the station prompt messages to stations KEYPNCH and VERIFY.

```
SEND M=SCNP11B,LT=KEYPNCH
SEND M=SCNP11B,LT=VERIFY
```

Send cross-station notification only for stations which are printers.

```
SEND M=SPOC30,LT=*PRNTR*
```

Send station notification of job completion to station OUTPUT which is not a printer. This message should be suppressed for all other stations unless they are a printer. In this case, the SEND macros must be specified in the order below:

```
SEND M=SPRG30,LT=OUTPUT
SEND M=SPRG30,LT=*PRNTR*
```

9.7.4 CA-Driver Customization

Options set in the load module CA7AGENB determine date formatting in CA-Driver. The default option is to use American style date formats (such as MM/DD/YY). If CA-Driver is to use European style date formats, you must apply USERMOD UL23336. This USERMOD is supplied on SAMPJCL. To change the date option, change the DATE= keyword value on the CAI7GENB macro from MMDD to DDMM. Then follow the instructions to reassemble and link edit this module.

9.7.5 Reserved DDname Table - SASSPMDD

SASSPMDD is a table of reserved ddnames which is examined during the CA-7 database LOAD process to determine if a data set is to be flagged as **permanent (PERM)** when the data set is added to the database. See the discussion of the DB.6 screen in the *CA-7 Database Maintenance Guide* for information on permanent (PERM) data sets.

Data sets whose ddnames exactly match entries in SASSPMDD are flagged as **permanent (PERM)** when added to the CA-7 database. Generics are not supported.

The table SASSPMDD is distributed in source and load module format. As it is distributed, the table contains entries for such ddnames as STEPLIB, JOBLIB, STEPCAT, and JOBCAT. Additional entries may be added to the table to suit specific installation needs. To modify the table, change the source for SASSPMDD making sure that any updates conform to the existing table format as described in the module. After making appropriate source module changes, reassemble and link edit SASSPMDD. Because SASSPMDD is used by module SASSJCL during the LOAD process, ensure that it resides on every library where SASSJCL is found. See USERMOD UL23306 for a sample of an SMP update to the supplied SASSPMDD module.

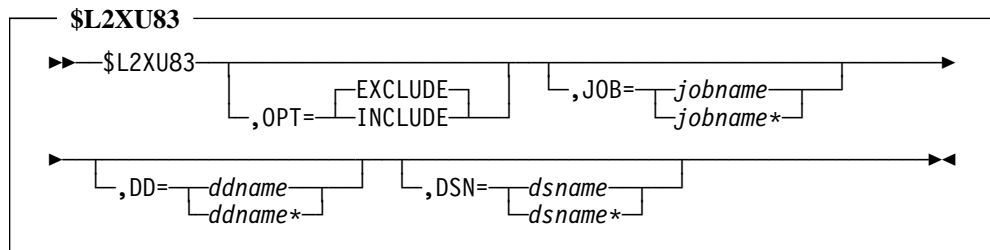
9.7.6 SMF Type 14/15 Record Exclusion/Inclusion Table

SASSXU83 is a table which can be used by the CA-7 U83 SMF interface to determine which type 14 and 15 records should be passed to the CA-7 address space through the communications data set. The table can contain specific or generic entries for job names, ddnames and/or data set names. The default usage of the table is to identify those 14/15 records which should be excluded from processing. However, you can override this to indicate the table should be inclusive (that is, process **only** those 14/15 records which match an entry in the table). The entries in the SASSXU83 are created using the \$L2XU83 macro.

9.7.6.1 Usage Notes

- If the SMF type 15 record is not passed to CA-7, data set triggering cannot occur for the data set.
- Reducing the volume of SMF data passed to CA-7 can significantly improve overall performance.
- If you wish to suppress ALL type 14 and/or 15 records, you may do so by setting options in ICMDSECT. CA-7 is distributed with the default settings to collect SMF type 15 records but *not* to collect SMF type 14 records. See 10.2.2, “SMF Processing” on page 10-4.

9.7.6.2 \$L2XU83 Macro



Where:

OPT

Indicates whether SASSXU83 will be processed as an inclusion or exclusion table. If OPT is coded, it must be the first occurrence of \$L2XU83 in the module. If INCLUDE is specified, only the SMF 14/15 records which match criteria in the SASSXU83 table will be collected and passed to CA-7.

Size/Type: EXCLUDE or INCLUDE
 Default: EXCLUDE
 Required: No (mutually exclusive with other parameters)

JOB

Specifies a job name pattern to be matched with SMF 14/15 records.

Size/Type: 1 to 8 alphanumeric characters (may include ? and/or * mask characters)

Required: No (mutually exclusive with other parameters)

DD

Specifies a ddname pattern to be matched with SMF 14/15 records.

Size/Type: 1 to 8 alphanumeric characters (may include ? and/or * mask characters)

Required: No (mutually exclusive with other parameters)

DSN

Specifies a data set name pattern to be matched with SMF 14/15 records.

Size/Type: 1 to 44 alphanumeric characters (may include ? and/or * mask characters)

Required: No (mutually exclusive with other parameters)

9.7.6.3 Pattern Masking

The JOB=, DD= and DSN= keywords support mask patterns with wildcard mask characters (?) and generic mask characters (*). These may be used anywhere in the pattern mask. If the pattern you specify does not contain any ? or * mask characters, the target string will have to match the pattern EXACTLY to satisfy the comparison.

Wildcard masks (?) will match any single character in the target string. For example, a mask of 'A?C' will match on 'ABC', 'AXC', 'A7C'; but, it will not match on 'ACX', or 'AC'. If you are looking for any four character string which begins with A and ends with C, specify the pattern 'A??C'.

Generic masks (*) will match a string of characters of any length (including zero). For example, a mask of '*ABC' will match on any string that ends with the characters ABC (including 'ABC' with no prefix). When used with the DSN= keyword, an embedded generic mask will only have a scope of the current data set node. For example, a mask of 'ABC.*.XYZ' will match any data set name which has EXACTLY three nodes where the first is 'ABC' and the third is 'XYZ'. Generic masks at the end of the pattern have an unlimited scope. That is, mask 'ABC.*' will match on any data set name that begins with 'ABC.' regardless of the number of nodes. Data sets 'ABC.Z' and 'ABC.W1.X2.Y3.Z4' would both match the mask.

Note: Be very careful when using wildcards and embedded generics in your patterns. Carefully think through what it is you are trying to match on and make the pattern AS SIMPLE AS POSSIBLE. If you do not keep this in mind, you will probably end up excluding or including data sets you did not intend to match on.

9.7.6.4 Defining the SASSXU83 Module

The SASSXU83 module is comprised of one or more \$L2XU83 macro calls. After the module is coded, it must be assembled and linked. See the CA-7 Sample JCL library for sample JCL to receive and apply SASSXU83 to CA-7 as a USERMOD. See member \$\$INDEX which contains an index of the members in SAMPJCL.

The SASSXU83 module must be accessible to CAIRIM when CA-7 is initialized by the CAS9 procedure. For example, it can be placed in the same library containing the SASSU83 module used by CAIRIM. If the table needs to be modified after it is loaded, CAIRIM can be executed with a PARM(REINIT) for CA-7 to refresh the CSA copy.

SASSXU83 Example

	column	
	10 16	72
	TITLE 'CA-7 SMF 14/15 FILTER TABLE'	
SASSXU83	START	
	PRINT ON	
	\$L2XU83 OPT=EXCLUDE	
	\$L2XU83 JOB=ABC	
	\$L2XU83 JOB=ABCDE*	
	\$L2XU83 DSN=MY.COMMON.PARMLIB	
	\$L2XU83 DSN=PAY.*	
	\$L2XU83 DSN=PROD.*.*DBASE	
	\$L2XU83 DSN=*.???QUE.PROD	
	END	

9.8 Function Aliases for Formatted Screens

All function values for formatted screens may be assigned alternate names. This allows for abbreviation (for example, L for LIST). It also permits alternate values such as CHANGE for UPD. The only restrictions are that the alias names must be 8 characters or less, and must not conflict with a function or other alternate name. The following alias names are distributed with the system:

Function(s)	Service level	Alias
ADD	ADD	A,ADDT,AELETE,AIST,APD
APPEND	READ	AP,APP
APPENDP	READ	n/a
CLEAR	n/a	CL,CLR
DD	DELETE	n/a
DELETE	DELETE	D,DEL,DELT
DELPRRN	UPDATE	n/a
EDIT	n/a	E,EDITH
EXIT	n/a	n/a
FE	READ	FEIT,FEPL,FEVE
FETCH	READ	F
FETCHP	READ	FP
FORMAT	n/a	FMT,FOR,FORM
FPE	READ	n/a
FREE	DELETE	n/a
LIST	READ	L,LDD,LDIT,LISTA,LISTP,LISTR,LPD
RENAME	UPDATE	REN
REPL	UPDATE	R,REP
REQ	UPDATE	n/a
RESOLV	SUBMIT	RES
RET	SUBMIT	n/a
RUN	SUBMIT	n/a
RUNH	SUBMIT	n/a
SAVE	ADD	S
SR	UPDATE	n/a
SS	ADD	n/a
SUBMIT	SUBMIT	SUB
UPD	UPDATE	U,UDD,UIST,UPDATE,UPDT
XPOST	UPDATE	n/a
XPRE	UPDATE	n/a
XQ	UPDATE	n/a
XQJ	UPDATE	n/a
XQM	UPDATE	n/a
XQN	UPDATE	n/a
XRQ	UPDATE	n/a
XRST	UPDATE	n/a
XSPPOST	UPDATE	n/a

Function(s)	Service level	Alias
XSPRE	UPDATE	n/a
XUPD	UPDATE	n/a
XWLP	UPDATE	n/a

Chapter 10. Performance and Tuning

This chapter discusses some steps that can be taken to fine tune CA-7 and achieve a much more efficient operation.

10.1 Monitoring CA-7 Performance

There are a variety of methods available for monitoring CA-7's performance as well as monitoring the processing of those jobs that CA-7 submits. These methods are described below.

10.1.1 Automated Performance Analysis (APA)

The APA system provides a facility for viewing up-to-the-minute information about many aspects of CA-7 processing. The data is displayed using GRAPHx commands which indicate the type of information to be shown. Some of the common graphs for analyzing performance include:

```
GRAPHS,ID=190 - Percentage of Transactions with Response Time Less than 10 Seconds
GRAPHS,ID=750 - Communication Dataset Reads Vs. Busy Conditions
GRAPHS,ID=580 - CA-7 UP Time Vs. SMF Task Time
GRAPHS,ID=530 - CA-7 Up Time Vs. Job LOAD Time
GRAPHJ,ID=1190 - Percentage of Jobs Scheduled Via Schedule Scan
```

See the *CA-7 Reports Guide* for additional information about the APA facility.

10.1.2 CA-Earl Reports

CA-Earl is an easy-to-use report writer which can be used to produce reports from CA-7 log data or database backup data. Several reports are supplied with CA-7. These can be used as supplied or modified to meet installation needs. The following reports may be useful when analyzing performance:

```
CA7ER028 - Job Termination Posting Dwell Time
CA7ER029 - Job Completion Dwell Time
CA7ER031 - Transaction Response Time Profile
CA7ER035 - Performance Statistics Information Job Report
CA7ER036 - Performance Statistics Information System Report
```

See the *CA-7 Reports Guide* for additional information about the CA-Earl reporting facility.

10.1.3 History Reporting

The CA-7 history reporting facility uses CA-7 log records from the log history and log archive files to produce historical reports. The following reports may be useful when analyzing performance:

<pre>06 - Job Processing Activity Report 70 - Internal Activity Trace Report</pre>
--

See the *CA-7 Reports Guide* for additional information about history reporting.

10.1.4 /DISPLAY Command

The /DISPLAY command can also be useful in gathering information about CA-7 performance and environment. The following options may be useful:

<pre>/DISPLAY,DB=ALL - To list information about the database files /DISPLAY,PL=ALL - To list information about CA-7 internal main memory storage pools /DISPLAY,Q=ALL - To list allocation information about CA-7 queue files /DISPLAY,A=ALL - To list information about usage of CA-7 application programs</pre>
--

See the *CA-7 Commands Guide* for additional information about the /DISPLAY command.

10.2 Initialization File Considerations

The following are some considerations related to optimizing performance of CA-7.

10.2.1 Schedule Scan Parameters

When specifying time span and increments for schedule scan wake-up and queue loading, shorter time spans and more frequent increments place less load on the system each time that a wake-up occurs. Since fewer jobs are placed in the request queue, job completion processing is also improved. See the initialization file SCHEDULE statement parameters SCNINCR and SCNSPAN. (See 5.2.21, “SCHEDULE Statement” on page 5-50.) An increment of 2 and a span of 4 are good default values.

10.2.2 SMF Processing

The PERFORM parameter of the INIT statement has options which can reduce database access during the SMF feedback process. Options 1 and 6 address this area of performance. (See 5.2.4, “INIT Statement” on page 5-13.)

There is also a way to cause CA-7 not to collect any SMF type 14 (x'0E') records. These records are produced when a data set is closed after being opened for input. If the PERFORM=1 option is being used, then CA-7 is ignoring these SMF records but they are still being collected by the CA-7 SVC and ICOM. To eliminate the collection by the SVC, a zap can be applied to module ICMDSECT. After the zap is applied, the modified load module must be copied to the CAILIB and then a COLD reinitialization using CAIRIM must be done (or an IPL).

NAME	ICMDSECT	ICMDSECT
VER	0005	00
REP	0005	02

There is also a way to cause CA-7 not to collect any SMF type 15 (x'0F') records. These records are produced when a data set is closed after being opened for output. This option can also be set by zap in ICMDSECT as described above. The option is in the same byte (offset +05) and the bit is (x'04'). If the bit is turned on, NO SMF 15 RECORDS ARE COLLECTED. Use this option with caution since it effectively disables CA-7's ability to perform data set triggers or automatically satisfy data set requirements.

The volume of SMF 14/15 records processed by CA-7 can also be reduced by using the SASSXU83 table module. This table allows SMF 14/15 records to be excluded based on job name, ddname, and/or data set name. Alternately, it can be structured to only include records based on job name, ddname, and/or data set name. See 9.7.6, “SMF Type 14/15 Record Exclusion/Inclusion Table” on page 9-54 for specific details. Check member UL233IZ in SAMPJCL for example of SMP USERMOD.

10.2.3 Queue IOB Count

It is recommended that at least 5 be specified as the value for the initialization file DAIO statement IOB parameter. (This is the value used in the Stage 2 JCL generated at SYSGEN time.) As the number of terminals accessing the queues increases, this value should increase by 1 (or more) for every 10 terminals. However, a value over 9 should never be necessary.

When the option to make some queues memory-resident is taken, the value of 5 may be sufficient for an even larger number of terminals. See the initialization file FORMAT statement parameter INDEX. (See 5.2.26, “FORMAT Statement” on page 5-63.)

10.2.4 Application Pool Sizes

The size of the application pool should never be less than 80K. Size should be increased if many different functions of CA-7 are to be used concurrently. Value should never exceed 120K. See the initialization file RESIDENT statement APGPL parameter. (See 5.2.1, “RESIDENT Statement” on page 5-10.) Use the /DISPLAY, POOLS=ALL and the /DISPLAY,A=ALL commands to monitor pool size.

It may also be desirable to add APPLCTN statements to the initialization for CA-7 programs which are highly used. The /DISPLAY,A=ALL command can be used to monitor program loads and to determine if some programs should be marked ATTR=PERM so that they are only loaded once during CA-7 initialization.

10.2.5 Terminal Dispatching Priority

CA-7 dispatches terminals for service based on how the GROUP, LINE, and TERM statements are coded in the initialization file. Care in the use of the statements can favorably impact terminal performance.

GROUP, LINE, and TERM statements have certain restrictions on how they are grouped together. However, within those restrictions, the earlier that a LINE statement appears within the initialization file, the higher the priority given to the terminals on that LINE.

The order in which terminal names appear within the TNAME value on each LINE statement further determines the relative priority of each terminal within that LINE group. Priority decreases in a left-to-right manner based on the sequence of the terminal names in the TNAME list. Printers should be placed at the end of their line; the trailer terminal should be defined as the last terminal in the initialization file; batch terminals should be defined just before the trailer terminal; TRX terminals should be defined just ahead of batch terminals; and CCI terminals should be defined before TRX terminals.

The total number of terminals that are defined also have an effect on performance. Because of this, use care in determining the number of terminals coded in the initialization file.

10.2.6 Memory-Resident Queues

In extremely large operations, when the work volume increases, a point may be reached in the CA-7 environment where the demand for queue access may equal or exceed the capabilities of DASD devices where the queues reside. In this event, the user has the option to move the most active CA-7 status queues (request, ready, active) into main memory residency. The memory-resident queues are placed in storage above 16M.

Selection of queues to be memory-resident is performed individually through the FORMAT statement INDEX option in the initialization file (see Chapter 5, "Initialization" on page 5-1). If the option is not taken, the queues remain on DASD as before. However, if the option is taken, the entire queue is brought into main memory at initialization time for CA-7. All input requests are serviced only from memory-resident queues; however, to ensure restartability, all output requests are still processed against the DASD data set also.

Experience with memory-resident queues may prove that the guidelines outlined here for placement of DASD queues could be relaxed or otherwise modified while still providing acceptable performance levels. The individual data center must make this determination based on its own experience and environment.

The memory required for these options is based on the allocated DASD space. Any reduction in the allocated DASD space correspondingly reduces the memory requirements.

Allocations can be monitored through the /DISPLAY command Q option and the CA-7 Queue Allocation Usage Profile report. The queue migration procedure in Appendix A, "Migrations" on page A-1 should be followed whenever queue allocations on DASD are changed.

Note: Never allocate only one track for any of the queue data sets.

See the "CA-Earl and CA-Easytrieve Plus Reporting" chapter in the *CA-7 Reports Guide* for details on the report available through CA-Earl request CA7ER027.

The trailer queue, though not a status queue, will be kept in memory when CA-7 is executing on OS/390, MVS/XA, or MVS/ESA systems. The entire queue is not loaded, but the individual data is loaded in memory as it is referenced. This enables CA-7 to keep the used portion of the trailer queue in memory without having to reserve storage for the size of the entire queue. The memory-resident trailer queue is placed in storage above 16M.

10.3 Calendar Schedules

You can significantly reduce schedule scan processing overhead by using trigger scheduling instead of calendar oriented schedules whenever possible. Using trigger scheduling causes jobs to enter the request queue in a more timely fashion and decreases the number of jobs in the queue at one time. Because of this, job completion performance improves. Also, fewer calendar schedules allow schedule scan to read fewer entries from the database to process the defined workload.

10.4 Operating System Considerations

10.4.1 Nonswappable

CA-7 and all ICOMs are nonswappable in OS/390 environments when they initialize.

10.4.2 Dispatching Priority

In systems running large online applications (TSO, CICS, IMS, and so forth), dispatching priority can significantly influence terminal response. If relatively few terminals are used for CA-7, allowing it to dispatch above the other online systems should yield better response while not likely to adversely affect the other online systems.

Note: The dispatching priority for ICOM will affect the processing performance of SMF feedback to CA-7.

10.5 Data Set Placement Recommendations

Take the following conditions into consideration when making the determination of where to place data sets. Base the decisions on your operating environment.

- Queues on a single pack
- Queues on multiple packs
- Queues on fixed-head disk drives
- Multiple CPU environment
- Database

In all cases, it helps to have data sets on packs with low-access activity by other systems or products.

10.5.1 Queues on a Single Pack

All queues on a pack should be adjacent to each other and as close to the VTOC as possible. If all the queues are on one pack, the preferred order is:

```

Active queue      )
Ready queue       )place next to each other
Request queue     )

DQTQ (place next to scratch queue)
Scratch queue (place next to DQTQ)

Trailer queue (place next to prior-run queue)
Prior-run queue (place next to trailer queue)

Postprocess queue
Preprocess queue

```

10.5.2 Queues on Multiple Packs

When queues are divided between packs, queues on a given pack should follow the above order. This order is based on experience with many sites. However, when full production is reached, you may want to examine EXCP counts and queue sizes for the last four queues. Additional APA graphs are available which provide activity counts. A different ordering may be optimum, if activity is atypical. The trailer queue and prior-run queue should always be next to each other, but their order can be experimentally reversed for fine tuning.

10.5.3 Queues on Fixed-Head Disk Drives

If your drives have a fixed-head portion, performance can be significantly improved by having the active queue, ready queue, and as much as possible of the request queue under the fixed head. When DQTQ is not on the same pack as the active, ready, and request queues, performance is enhanced by putting it under the fixed head.

10.5.4 Data Set Placement Restrictions

As with any online application, performance can be dramatically affected by the placement of the data sets which are used. Proper selection of type of storage and physical location can assure optimum performance. Conversely, poor selection can result in totally unsatisfactory performance.

Some of the user options discussed in previous paragraphs are specifically related to performance of access to data sets. More options are discussed in the following materials in addition to specific recommendations on the placement of CA-7 data sets.

Generally, it helps to have the data sets on packs with otherwise low-access activity from other systems. Whenever possible for queue data sets, making the queue(s) memory-resident provides much better performance and slightly lessens the importance of placement as outlined in the following discussions of queues. However, to ensure optimum performance, the placement guidelines should be followed whether or not memory-resident queues are used.

In many data centers, products that dump entire packs are used to back up data sets. These are highly efficient for this purpose; however, they sometimes monopolize channel activity during this process. To avoid performance lags in CA-7, its data sets should not reside on drives or channels which may be dominated by such products at a time when CA-7 performance is important. Any use of such products for backing up other data sets not related to CA-7 could also degrade the CA-7 performance if they are accessed through the same channel(s) used by CA-7. If such backups can be scheduled to occur during off-hours for CA-7, perhaps at the same time that the CA-7 backups are done, no problems should arise.

The following discussions address particular considerations for specific data sets or groups of data sets.

10.5.4.1 Communications Data Set

In a multi-CPU environment, this data set must be on a shared volume which is accessible to each CPU where CA-7 or ICOM will be executing. The volume should be accessed infrequently by other systems and should not contain CA-7 queue data sets, databases or logs if at all possible. In a multi-CPU environment, a reserve is issued against this data set. This must be considered when deciding where to locate the data set to minimize potential performance problems which could result.

The communications data set is accessed by CA-7 and any ICOMs that are executing. The reserve that is done should be excluded from the control of any software package that converts reserves to global ENQs (for example, GRS or MIM). The QNAME is UCC7CMDS and the RNAME is CMDS.

Among data sets that should not be on the same pack are:

TMC for CA-1

CMT for CA-11

JEHF for CA-11

SPOOL pack, page pack

Any high-access data sets for another online system (TSO, CA-Roscoe, CICS, and so forth).

10.5.4.2 Checkpoint Data Set

If this data set cannot be separated from the CA-7 queue data set volume(s), it should be allocated adjacent to UCC7DQTQ and UCC7SCRQ data sets.

10.5.4.3 CA-7 Load Library

This load module library should be located on a low-access drive whenever possible.

10.5.4.4 Placement of Queue Data Sets

Some general performance considerations apply to queue data sets just as they would for any data set which is critical to an online system. For example:

- Using drives with little or no contention from other systems.
- Avoiding domination of devices and/or channels by products which dump entire packs.
- Fixed-head access is significantly faster.

With the exception of queues under a fixed head, all queues on a pack should be adjacent to each other and as close to the VTOC as possible.

Table 10-1. Multiple CPU Environment					
Available packs					
	1	2	3	4	5
Using 2 drives	Communi- cations data set	All others			
Using 3 drives	Communi- cations data set	Index data set Dataset data set	All others		
Using 4 drives	Communi- cations data set	Index data set Dataset data set	Active queue DQTQ	All others	
Using 5 drives	Communi- cations data set	Index data set Dataset data set	Active queue	DQTQ	All others
Note: The status queues can also be memory-resident for large volume users where DASD access time is critical and of prime consideration.					

Table 10-2. Single CPU Environment				
Available packs				
	1	2	3	4
Using 2 drives	Index data set Dataset data set	All others		
Using 3 drives	Index data set Dataset data set	DQTQ	All others	
Using 4 drives	Index data set Dataset data set	DQTQ	Active queue	All others
Note: The status queues can also be memory-resident for large volume users where DASD access time is critical and of prime consideration.				

10.6 Database Controls

The UCC7DBASE statements in the DBPARMS member of the CA-7 JCLLIB provide the parameters for database access.

The CA-7-unique parameters in the UCC7DBASE statements, STRINGS, BUFNI, and BUFND, can be varied by the user to attain whatever performance levels are desired.

Note: The STRINGS parameter must never be set to 1 for any data type.

The BUFNI and BUFND parameter values should be the selected STRINGS parameter value + 5.

The /DISPLAY,DB=ALL command can be used to display current STRINGS and BUFNI/BUFND values.

Additionally, the IDCAMS parameters, specified with the IDCAMSDEFINE keyword, offer full support for VSAM performance options just as if CA-7 were not involved. The VSAM options for CI and record sizes, IMBED, REPLICATE, buffer space, and so forth, can be varied as necessary to attain the desired overall performance level. Those parameters are governed only by the constraints imposed by VSAM itself.

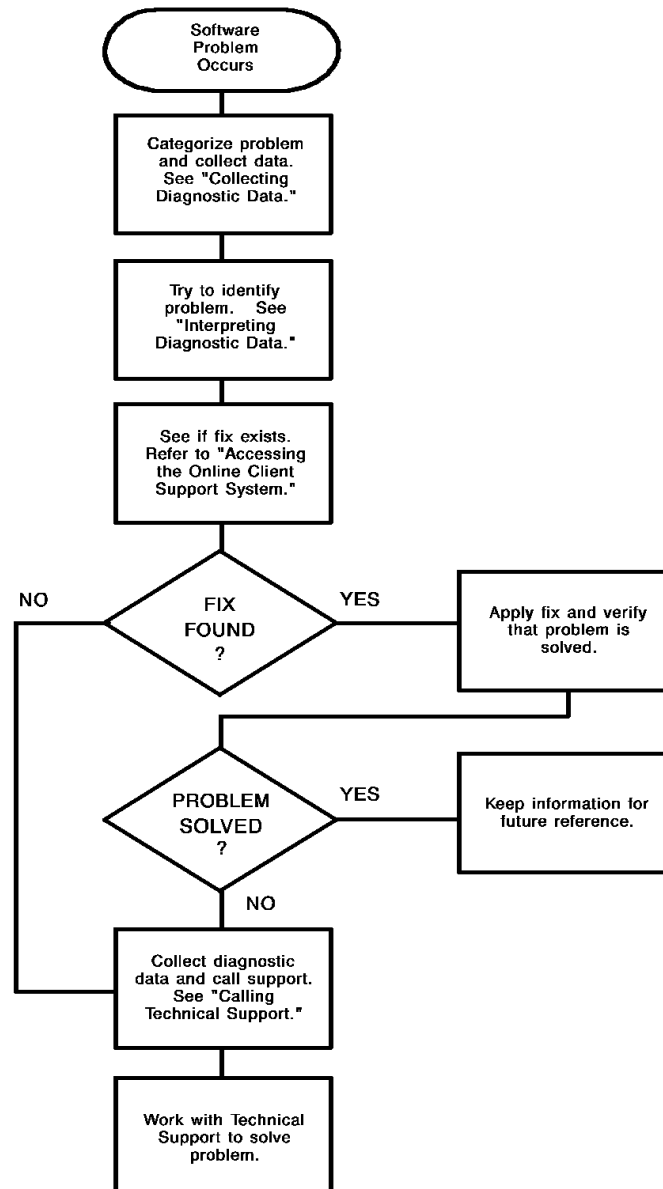
Chapter 11. Troubleshooting

This chapter contains information about:

- Identifying and resolving problems
- Contacting Computer Associates Technical Support
- Receiving ongoing product releases and maintenance
- Requesting product enhancements

11.1 Diagnostic Procedures

Refer to the flowchart below for a summary of the procedures you should follow if you have a problem with a Computer Associates software product. Each of these procedures is detailed on the following pages.



11.1.1 Collecting Diagnostic Data

The following information is helpful in diagnosing problems that might occur:

- Control statements used to activate your product
- JCL used to install or activate your product
- Relevant system log or console listings
- Relevant system dumps or product dumps
- List of other IBM or third-party products that might be involved
- Manufacturer, model number, and capacity of your hardware
- Numbers and text of IBM or CA error messages associated with the problem
- Names of panels where the problem occurs
- Listings of all fixes applied to all relevant software, including:
 - The dates fixes were applied
 - Fix numbers
 - Names of components to which fixes were applied
- Short description of problems

11.1.2 Interpreting Diagnostic Data

When you have collected the specified diagnostic data, write down your answers to the following questions:

1. What was the sequence of events prior to the error condition?
2. What circumstances existed when the problem occurred and what action did you take?
3. Has this situation occurred before? What was different then?
4. Did the problem occur after a particular PTF was applied or after a new release of the software was installed?
5. Have you recently installed a new release of the operating system?
6. Has the hardware configuration (tape drives, disk drives, and so forth) changed?

From your response to these questions and the diagnostic data, try to identify the cause and resolve the problem.

11.2 Accessing the Online Client Support System

Computer Associates is making extensive use of the Internet for your benefit. CA encourages you to "surf the net" to the CA home page at <http://www.cai.com>. The CA Internet site provides a great variety of information about CA products and services, including:

- Service and support
- Product information and sales
- CA-World conference information
- Press releases
- CA user groups

CA-TCC (CA-Total Client Care) gives you real time, interactive access to CA product support information through the Internet. Using CA-TCC, you can:

- Open new issues
- Browse or update your existing issues
- Perform keyword searches
- Download solutions, PTFs, and important notices regarding CA products, maintenance, and documentation

11.2.1 Requirements for Using CA-TCC

The following are the requirements to use CA-TCC:

- You must be a CA client with a current maintenance agreement.
- You must register through the CA Internet site.
- You must access the Internet with a browser that supports the HTML specification 2.0 or higher, such as Netscape Navigator 2.0 or higher or Microsoft Internet Explorer 3.0 or higher.

Browsers that meet the HTML requirement support the following functions, which are required for CA-TCC:

- Secure sockets layer (SSL) to encrypt your transaction traffic
- Encrypted data records (known as COOKIES)
- HTML tables

11.2.2 CA-TCC Security

CA-TCC runs as a secured server (SSL). You may need to configure your browser to enable SSL. Guidelines for doing this are provided on the CA Technical Support page.

11.2.3 Accessing CA-TCC

To access CA-TCC, click on Support on the CA home page and follow the links for CA-TCC. The CA-TCC options are:

- CA-TCC Information
- CA-TCC Registration
- Access CA-TCC

These options are described below.

CA-TCC Information: Select the information option to view background information for CA-TCC, details about the prerequisites, and instructions for configuring your browser. Be sure to review this section for updates or information not included here.

CA-TCC Registration: Select the registration option to identify yourself to CA-TCC. You must register before you can access CA-TCC online. There are prompts for all required information, including your name, site ID, CA-StarTrak PIN, company name, E-Mail address, postal address, and desired password for accessing CA-TCC.

Note: If you do not have a CA-StarTrak PIN, CA-TCC provides one for you when you register.

Access CA-TCC: Select the access option to begin using CA-TCC. When prompted, enter your user ID and password. Once your sign-on is validated, you can select one of the following options:

Open a New Issue

Open an issue for, or request an enhancement to, one of your CA products.

Browse Your Issues

Display all issues for your site. The issues are grouped into three categories: Open, Closed, and Enhancement Requests (DARs).

Browse/Download Solutions

Specify criteria for selecting solutions, which you can then view or download.

Search CA Knowledge Base

Specify criteria for searching the CA database for solutions, problems, and keywords that can provide you with immediate answers to your product support questions and concerns.

Update Your CA-TCC Profile

Make changes to your default E-mail address, phone number, and password whenever necessary.

Display Your Site's Licenses

View a list of all the CA products for which your company site is currently licensed.

Display News Items

View and download recently published solutions for CA products, instructions for downloading from CA-TCC, and helpful information for using CA-StarTrak, CA-TCC, or other CA products.

11.2.4 Accessing the Technical Support Phone Services Directory

The Computer Associates Technical Support Phone Services Directory lists each CA product and the telephone number to call for primary support for that product. To access the Support Phone Services Directory online, click on Support on the CA home page. Follow the links, first to CA Telephone Support and then to the Technical Support Phone Numbers directory.

11.2.5 CA-TCC Hotline

If you experience any problems using CA-TCC, please call the CA-TCC Technical Support hotline at 609-273-3412.

11.3 CA-TLC: Total License Care

Many CA software solutions use license keys or authorization codes to validate your hardware configuration. If you need assistance obtaining a license key or authorization code, contact the CA-TLC: Total License Care group at 1-800-338-6720.

11.4 Calling Technical Support

Computer Associates provides telephone support for all its products.

If you are in North America, refer to the *Technical Support Phone Services Directory* for the appropriate phone number. Outside North America, call your local Computer Associates Support Center during normal business hours.

Note: Only your local Computer Associates Support Center can provide native language assistance. Please use English when contacting any North American center.

If you are unable to locate the Technical Support phone number you need, call 1-800-645-3042 for assistance if you are in North America or 631-342-4683 outside North America. The operator will record your call and a Technical Support representative will call you back. After hours calls should be limited to severity 1 problems.

If you are unable to resolve the problem, please have the following information ready before contacting Computer Associates Technical Support:

- All the diagnostic information described in 11.1.1, "Collecting Diagnostic Data" on page 11-3
- Product name, release number, operating system and genlevel.
- Product name and release number of any other software you suspect is involved.
- Release level and PUTLEVEL of the operating system.
- Your name, telephone number and extension (if any).
- Your company name.
- Your site ID.
- A severity code. This is a number (from 1 to 4) that you assign to the problem. Use the following to determine the severity of the problem:
 - 1 a "system down" or inoperative condition
 - 2 a suspected high-impact condition associated with the product
 - 3 a question concerning product performance or an intermittent low-impact condition associated with the product
 - 4 a question concerning general product utilization or implementation

11.5 Product Releases and Maintenance

Clients are requested to operate only under currently supported releases of the product.

Clients with current maintenance agreements also receive ongoing product maintenance. When a new release of the system is available, a notice is sent to all current clients.

11.6 Requesting Enhancements

Computer Associates welcomes your suggestions for product enhancements. All suggestions are considered and acknowledged. You can use either of two methods to request enhancements:

- Contact your Account Manager who will initiate a Demand Analysis Request (DAR) for you.
- Enter your request through StarTCC Extended Support on the Web.

Appendix A. Migrations

This appendix discusses the steps required to convert from one database organization to another and the steps required to migrate CA-7 queues.

A.1 Migrating Queues to New Devices

Queue Migration allows the user to expand, reduce or relocate queue data sets without losing queue data. Relocation can be to another device of the same type or to a different device type. Because queue data is transient, this procedure cannot be used for backup or recovery. No CA-7 activity can occur between the unload and reload of the queues.

The procedure for queue migration consists of the following steps:

- Step 1 -** Reallocate the queue data sets as desired (with new data set names). All queues and the checkpoint data set must be on the same device type.
- Step 2 -** Unload the queues using /SHUTDOWN with a DMPQ or Z4 or Z5 option. Any current log data is lost when Step 6 is performed. The current log can be saved by running the log dump job.
- Step 3 -** Verify the WTO CA-7.937 was issued indicating successful dump of queues. If not successful, make the necessary changes, reinitialize CA-7 with TYPE=ERST, and perform Step 2 again.
- Step 4 -** Change the queue definitions in the CA-7 JCL and catalog the new queues.
- Step 5 -** If reloading to a different device type, change the DASD and RECORDS parameters in the DAIO statement.
- Step 6 -** Reinitialize CA-7 with the MOVQ startup option.

You may use the following programs to accomplish these procedures:

- SASSINQD - Unload the queues
- SASSINQB - Reload the queues

A.1.1 Unloading the Queues

Use the DMPQ or Z4 or Z5 parameter of the /SHUTDOWN command to trigger the unloading of the queues. A Queue Dump data set (ddname UCC7QDMP) must be allocated and be available in the CA-7 online execution JCL. See 6.3.1.4, “MOVQ Start (Move Queues Restart)” on page 6-21 for more information concerning /SHUTDOWN,DMPQ. The Data Set Requirements chapter discusses the Queue Dump data set in more detail. (See 3.2.5.6, “Queue Dump Data Set” on page 3-18.)

The SASSINQD module receives control during shutdown whenever a DMPQ type of shutdown is entered and unloads the CA-7 queues. Once the queues have been successfully dumped, CA-7 continues normal shutdown processing, depending on shutdown options.

A user exit (SASSXXQD) is provided for the user to look at, modify, delete, or add records going to the Dump file. See 9.3.19, “Dump Queue” on page 9-31 for more information about this exit.

The CA-7 queues are unloaded in the following sequence:

1. Preprocess queue
2. Request queue
3. Ready queue
4. Active queue
5. Postprocess queue
6. Prior-run queue

Data in the trailer queue is also unloaded during the dump of the above queues. The CA-7 scratch queue which contains messages, and the DQTQ are not unloaded. However, this same procedure must be used if moving or reallocating the scratch queue or DQTQ.

For each queue, segments are unloaded in the following sequence:

1. Job/station headers
2. JCL segment trailers (for jobs only)
3. Input requirement trailers (for jobs only)
4. Output requirement trailers (for jobs only)

A.1.1.1 Virtual Resource Management Considerations

The VRM database component contains "active" resource records which relate to jobs currently in the CA-7 queues. The VRM active resource records track resource use by job and if in use must match the CA-7 queue records. These active records are dumped to the VRM Dump queue file (ddname CA7VDMP) and are date/time stamped to match the queue record(s) to ensure synchronization during DMPQ/MOVQ processing.

A.1.2 Reloading the Queues

To reload the queues after performing a /SHUTDOWN,DMPQ, specify TYPE=MOVQ as the PARM input on the EXEC JCL statement for CA-7. (This approach overrides the TYPE parameter specified in the initialization file.)

If a different device type is to receive the data that was unloaded, remember to change the DASD parameter in the DAIO statement in the initialization file to reflect the new device type.

The SASSINQB module receives control whenever TYPE=MOVQ is specified. It then reloads the queues from the CA-7 Queue Dump file (ddname UCC7QDMP). This DD must define the same data set as that used to unload the queues.

When the MOVQ feature is selected, the appropriate DISP fields on the FORMAT cards in the initialization file are forced to F. Since the queues are formatted before the MOVQ takes place, a successful shutdown using the DMPQ parameter must have occurred prior to the startup.

The MOVQ parameter is identical to a TYPE=FORM in the initialization process except the CA-7 queues are reloaded.

A.1.2.1 Virtual Resource Management Considerations

During the reload process, the VRM Dump file is read to determine if any VRM active resource records must be reloaded to match the CA-7 queue records. The VRM active records must have been unloaded during the same DMPQ shutdown process as the queue records. A control record is written to the VRM Dump file to ensure that the queue records and the VRM active records are synchronous. Any VRM "active" records found on the VRM database component are deleted prior to reloading the active records found on the VRM dump file.

Index

Special Characters

\$\$INDEX member 9-2
\$\$INDEX SAMPJCL member 9-3
\$L2EXTT macro 3-73
\$L2XDSN macro 3-79
\$RXEQ JES command 5-52

A

Access restrictions 1-10
Activating/deactivating workload balancing
 changing selection parameters 3-69
 LWLB 3-69
 scheduling WLB modules 3-69
 WLB 3-69
Active queue
 overview 2-11, 2-14
 space requirements with external tasks 3-70
Aliases
 command 9-18
 table 9-57
ALOG1 statement 5-46, 6-12
ALOG2 statement 5-46, 6-12
Alternate terminal 6-25
APA reporting 1-13, 9-44, 10-2
APPLCTN statement 5-22
Application
 modules, defining 5-17
 pool sizes 5-10, 10-5
 programs 2-5
Archive data set
 file requirements 3-25
 overview 2-16
Archive JCL example 8-6
Archives Purge
 program (SASSHIS6) 8-9
 report 8-13
ARF (Automated Recovery Facility)
 and recovery 7-25
 CA7ARK procedure 7-25
 CA7ARL procedure 7-25
 data set
 overview 2-9
 space requirements 3-8
IDCAMS define parameters 7-27
overview 1-9

Assigning PF/PA keys 5-82, 5-83
Automated Performance Analysis (APA) reporting 1-13
Automated Production Control system 1-6

B

Backup and recovery
 database
 backup DD statements 7-7
 backup procedures 7-2
 backup/reload JCL 7-6
 reload procedures 7-11
 reports 7-20
 UCC7DBASE statements 7-16
 recovery aid facility, reporting 7-37
 terminals
 batch terminals 7-34
 VTAM terminals 7-33
 workload considerations
 CA-7 7-35
 other CA products 7-35
 other vendor software 7-35
Backup, VRM 7-28
Backup/Reload
 Control report (SASSBK00-01) 7-20
 database 2-10
 Database Inventory report (SASSBK07-01) 7-22
Base calendars
 CALBLK statement 5-26
 overview 2-5
Batch
 access 2-19
 job vs. started task 6-16
 terminal recovery 7-34
Batch Card Load Program (BCLP) 1-11
Batch Card Load program exit 9-33
Batch support programs
 Backup/Reload 2-10
 Database Verification 2-10
Batch terminal data sets
 file requirements 3-19
 overview 2-16
 space requirements 3-19
BCLP
 and DBPARMS deck 7-15
 executing 2-19
 exit 9-33
 overview 1-11

- Broadcast messages 5-12
- Browse data set
 - BSAM terminal 2-20
 - overview 2-17
 - space requirements 3-21
- Browse Message exit 9-28
- BSAM terminal (browse data set) 2-20
- BTI
 - message table 9-49
 - program 2-19
- Buffer pool sizes 5-10

C

- CA-11
 - and RESTART statement 5-47
 - considerations 1-10
- CA-7 calendar PDS
 - overview 2-17
 - space requirements 3-24
- CA-7 Console Terminal Output exit 9-26
- CA-7 Cross-Platform Checkpoint File
 - overview 2-17
- CA-7 Cross-Platform Scheduling Checkpoint
 - file requirements 3-26
- CA-7 Cross-Platform Scheduling Profile
 - file requirements 3-25
 - overview 2-17
 - space requirements 3-25
- CA-ASM2
 - and backup 7-4
 - and recovery 7-12
- CA-Driver 1-11
 - customization 9-53
 - procedure library 3-22, 6-9
 - PROCLIB 2-17
 - variable substitutions 5-74
- CA-Earl reporting 1-13
- CA-Earl reports
 - performance reports 10-2
 - testing after installation 4-7
- CA-Easytrieve Plus reporting 1-13
- CA-JCLCheck
 - interface
 - JCLCHECK statement 5-61
 - options data set 3-22
 - REGION values 6-2
- CA-Netman
 - API deferred request data set
 - overview 2-17
 - space requirements 3-23
 - interface 9-37
- CA-Netman (*continued*)
 - NETMAN statement 5-71
 - transaction file 2-17
- CA-Netman interface 9-37
- CA-Panvalet interface 6-3
- CA-TCC (CA-Total Client Care) 11-5
- CA-TLC: Total License Care 11-8
- CA7ARK procedure 7-25
- CA7ARL procedure 7-25
- CA7LOAD 5-45
- CA7VBK program 7-28
- CA7VRL program 7-28
- CAIENF events 3-36
- CAIRIM 6-38
- CALBLK statement 5-26
- Calculating region size 6-2
- CALENDAR
 - macro 3-32, 5-26
 - statement 5-56
- Calendars
 - base, identifying 5-26
 - defining 3-32
 - definition and structure 3-30
 - example, base assembly listing 3-31
 - schedules 10-8
 - workstation 5-41
- CANCEL command, OS 6-25
- CANCEL statement 5-84
- CARAGENB load module 9-53
- CCI terminals 2-20
- Central Control System 2-4
- Chains, CSA/ECSA 6-44
- Changing
 - CA-Driver 9-53
 - function aliases 9-57
 - graphs 9-43
- Changing selection parameters
 - RESCHNG 3-69
 - WLB 3-69
 - XWLB 3-69
- Checkpoint data set
 - overview 2-16
 - placement restrictions 10-12
 - space requirements 3-13
- CI size 3-5, 7-19
- Class barriers 3-46
- CLBARR (WLB macro) 3-43, 3-65
- COLD
 - reinitialization 6-39
 - start (clear workload restart) 5-66, 6-20

- Command
 - aliases 9-18
 - exit 9-18
- Commands
 - DEMAND
 - cross-platform scheduling 5-20
 - demanding processing 2-12
 - DISPLAY 10-3
 - LOAD
 - and requirements 2-14
 - placing job in CA-7 database 2-12
 - LOADH 2-14
 - LWLB 3-69
 - MODIFY 6-30
 - OS CANCEL 6-25
 - RESCHNG 3-69
 - RUN
 - and cross-platform scheduling 5-20
 - and requirements 2-14
 - running a job independent of requirements 2-12
 - RUNDEF 5-20
 - SHUTDOWN 6-25
 - WLB 3-69
 - XWLB 3-69
- Communications data set
 - and ICOM 2-6
 - and INIT statement 6-11
 - and WARM start 6-18
 - CA-7 DD statements 6-6
 - file requirements 3-14
 - formatting 3-14
 - ICOM
 - DD statements 6-28
 - execution 6-32
 - overview 2-16
 - placement restrictions 10-12
 - space requirements 3-14
- Communicators, external
 - See* External communicators
- Comparison graphs 9-48
- Continuation formats 5-3
- Coupling Facility, ICOM 6-34
- CPM keyword 5-74
- CPU
 - multiple environment 5-54, 10-13
 - single environment 5-54, 10-13
 - statement 5-54, 6-12
 - usage 3-41
 - WLB macro 3-43, 3-62
- Cross reference of names in database
 - DSNAME-program 9-42
 - program-job 9-41
- Cross-platform scheduling
 - command usage 5-20
 - MONITOR 5-18
 - ROUTER 5-19
- CSA/ECSA chains 6-44
- CUST statement 5-12
- Customizing
 - CA-Driver 9-53
 - reports and graphs 5-12

D

- DAIO statement 5-69
- DASD definition, DAIO statement 5-69
- Data sets
 - additional 2-16
 - allocation 1-10
 - ARF data set 2-9
 - batch terminal data sets 2-16
 - browse data set 2-17
 - CA-7 calendar PDS 2-17
 - CA-7 Cross-Platform Checkpoint File 2-17
 - CA-7 Cross-Platform Scheduling Profile 2-17
 - CA-Driver PROCLIB 2-17
 - CA-Netman API deferred request data set 2-17
 - CA-Netman transaction file 2-17
 - Checkpoint data set 2-16
 - communications data set 2-16
 - dataset data set 2-8
 - Help Library 2-17
 - index data set 2-8
 - JCL data sets 2-10
 - JCLLIB 2-17
 - job data set 2-8
 - log data sets 2-16, 5-46
 - log history/archive data sets 2-16
 - options data set 2-17
 - override library 2-17
 - placement
 - recommendations 10-10
 - restrictions 10-11
 - queue dump data set 2-16
 - requirements summary 3-3
 - Statistics data set 2-16
 - submit data sets 2-16
 - tracking external 3-77
 - VRM data set 2-8
 - VRM dump data set 2-16
 - Workload Planning data sets 2-16
- Database
 - access 2-4
 - backup
 - and recovery 7-2

- Database (*continued*)
 - backup (*continued*)
 - DD statements 7-7
 - backup exit 9-38
 - backup/reload JCL
 - example 7-6
 - reload JCL considerations 7-14
 - cross-reference reports 9-40
 - files 3-6
 - Load Processing exit 9-35
 - maintaining 7-2
 - processing options 5-43
 - reload exit 9-40
 - space requirements 3-5
 - UCC7DBASE statements 7-16
 - verification 2-10, 7-15
- Dataset data set
 - description 2-8
 - space requirements 3-7
- Date and time validation option 6-15
- DB.1 screen 5-58
- DBASE statement 5-43
- DBPARMS data set 2-17, 7-15
- DCM 3-36
- DD statements 6-6
- DEFGRPH macro
 - defining comparison graphs 9-48
 - defining single graphs 9-48
 - syntax and examples 9-44
- Defining
 - calendars 3-32
 - holidays 3-33
 - months 3-34
 - terminals 5-33, 5-40
 - workstations 5-40
- Defining resources, RESIDENT statement 5-10
- Defining selection parameters 3-44
- Device definition 5-69
- DFDSS
 - and backup 7-5
 - and recovery 7-13
- DFTLTLB keyword 5-74
- Disaster recovery
 - planning 7-36
 - reporting 7-37
- Disk queue table (DQT) 2-11, 3-11
- Dispatching priority 10-9
- DISPLAY command 10-3
- DMPQ shutdown 5-78, 6-21
- Documentation
 - overview 1-12
 - recovery considerations 7-36

- DORM start 6-22
- Dormant CA-7 6-23
- Dormant CA-7 start 6-22
- DRIVER
 - variable substitutions 5-74
- DSNAME-program cross reference 9-42
- Dump Queue exit 9-31

E

- EDIT start 5-86, 6-22
- Emergency restart (ERST) 6-18, 6-19
- END statement 5-85
- Enhancements 1-1
- ENQ monitoring 6-24
- ENQ/RESERVE exit 9-13
- Environment 2-3
- ERST start 6-19
- Events, CAIENF 3-36
- Excluding SMF type 14/15 records 9-54, 10-4
- Execution
 - batch-only
 - initialization file 6-14
 - JCL 6-13
 - overview 6-11
 - ICOM 6-26
 - online
 - initialization file sample 6-10
 - JCL 6-4
 - overview 6-1
- Exits
 - Batch Card Load program (SASSBCX1) 9-33
 - Browse Message 9-28
 - CA-7 Console Terminal Output (SASSXX15) 9-26
 - coding 9-3
 - Command (SASSXX09) 9-18
 - Command (SASSXX10) 9-19
 - Command (SASSXX11) 9-21
 - Command (SASSXX12) 9-35
 - Database
 - Backup 9-38
 - Cross-reference 9-40
 - Reload 9-40
 - Database Load Processing (SASSXX12) 9-35
 - Dump Queue (SASSXXQD) 9-31
 - ENQ/RESERVE (SASSXX04) 9-13
 - External DSN Access (SASSXX07) 9-15
 - Forecast Worksheet (SASSXX08) 9-17
 - JCL
 - Attach Verification (SASSXX11) 9-21
 - Attach, example 9-47
 - Queue Entry (SASSXX05) 9-14
 - Submission (SASSXX02) 9-9

Exits (*continued*)

- Job Data Verification (SASSXX10) 9-19
- Job Name Verification (SASSXX01) 9-8
- Logoff (SASSXXFF) 9-30
- Logon/Password Verification (SASSXXLX) 9-29
- Move Queue (SASSXXQB) 9-32
- Personal Scheduling Verification (SASSXX14) 9-24
- SMF Feedback (SASSXX13) 9-22
- SMF WTO Message (SASSXX16) 9-27
- Utility Function Security Checking (SASSXX03) 9-11

External

- data sets, tracking 3-77
- DSN Access exit 9-15
- events 2-13
- tasks, tracking 3-70

External Communications Facilities 1-11

External communicators 2-19

Extracting log data 8-4

F

Flowcharts

- archives purge program 8-11
- history management program 8-5
- test network execution 4-6

FMTBLK statement 5-24

Forecast Worksheet exit 9-17

FORM start (Format Queues Restart) 6-22

FORMAT statement

- and batch execution 6-12
- and REGION values 6-2
- syntax and examples 5-63

G

Generic masks 9-55

Generic Unit Name/Device Type Table 9-51

GETMAIN macros 6-2

Graphs

- adding counters for new 9-47
- analyzing performance 10-2
- changing 9-43
- comparison 9-48
- JCL Attach Exit example 9-47
- single 9-48

GROUP statement

- changes required in batch mode 6-11
- syntax and examples 5-28

GRS (ENQ) 10-12

H

HELP facility 1-7

Help Library

- overview 2-17
- space requirements 3-21

History Management program (SASSHIS5)

- description 8-4
- reporting 7-36, 8-4

History records purge 8-9

History reporting 1-13

Holidays 3-33

I

ICOM

- activating 6-17
- and CPU statement 5-54
- and postinstallation testing 4-3
- cross-platform tracking 6-35
- dispatching priority 10-9
- execution 6-26
- initialization

- DD statements 6-28

- execution JCL 6-26

- ICOM Execution 6-26

MODIFY 6-30

overview 2-6

PARM values 6-27

usage considerations, general 6-29

WTOR replies 6-30

ICOM Coupling Facility 6-34

ICOM Coupling Facility Structure 3-27

IDCAMS

ALLVSAM

- backup 7-3

- recovery 7-11

and UCC7DBASE statements 7-16, 7-17

determining database space requirements 3-5

JCL 7-3

IDCAMSDEFINE parameters 7-19

IEHPROGM-type functions 1-10

Implementation steps 4-8

Including SMF type 14/15 records 9-54, 10-4

Incrementing with schedule scan 5-50

Index data set

- overview 2-8

- space requirements 3-7

INIT statement 5-13, 6-11

INIT2 statement 5-39

Initialization

- batch job vs. started task 6-16

Initialization (*continued*)

- COLD start 6-20
- configuration, host 5-13
- considerations 6-15
- date and time validation option 6-15
- DORM start 6-22
- EDIT start 6-22
- ERST start (Emergency Restart) 6-19
- FORM start 6-22
- ICOM 6-26
- log usage 6-16
- MOVQ start 6-21
- options 5-13
- see also Initialization file
- startup options 6-18
- WARM start 6-18

Initialization file

- ALOG1 statement 5-46
- ALOG2 statement 5-46
- APPLCTN statement 5-22
- application pool sizes 10-5
- CALBLK statement 5-26
- calendar schedules 10-8
- CALENDAR statement 5-56
- CANCEL statement 5-84
- contained in JCLLIB 2-17
- CPU statement 5-54
- CUST statement 5-12
- DAIO statement 5-69
- DBASE statement 5-43
- END statement 5-85
- FMTBLK statement 5-24
- FORMAT statement 5-63
- GROUP statement 5-28
- INIT statement 5-13
- INIT2 statement 5-39
- JCL statement 5-57
- JCLCHECK statement 5-61
- keywords 5-4, 5-5
- LINE statement 5-31
- NETMAN statement 5-71
- NEWS statement 5-12
- offline verification of
 - batch edit execution 5-86
 - edit errors 5-86
- OPTIONS statement 5-72
- performance considerations 10-4
- Pfnn and PAnn statements 5-83
- PFTERM statement 5-82
- queue IOB count 10-5
- queues, memory-resident 10-7
- RESIDENT statement 5-10

Initialization file (*continued*)

- RESTART statement 5-47
- schedule scan parameters 10-4
- SCHEDULE statement 5-50
- SECURITY statement 5-39
- startup procedures 6-17
- statements
 - sequence 5-5
 - syntax rules 5-2
- STATIONS statement 5-40
- STNCAL statement 5-41
- SVCNO statement 5-17
- TERM statement 5-33
- terminal dispatching priority 10-6
- termination 5-85
- UCC7VTAM statement 5-27
- VRMOPTS statement 5-81

Initiators 3-41, 3-45

INITR (WLB macro) 3-43, 3-63

Input network

- overview 2-12

Installation

- procedures 4-1
- test network execution 4-6
- testing 4-2

Installation requirements

- ARF data sets 3-8
- calendar
 - CALENDAR macro 3-32
 - definition and structure 3-30

data sets

- list 3-3
- requirements summary 3-3

database

- files 3-6
- space 3-5

dataset data set 3-7

features 3-2

file requirements

- batch terminal data set(s) 3-19
- browse data set 3-21
- CA-7 Calendar PDS 3-24
- CA-7 Cross-Platform Scheduling Checkpoint 3-26
- CA-7 Cross-Platform Scheduling Profile 3-25
- CA-Driver procedure library 3-22
- CA-Netman API deferred request data set 3-23
- checkpoint data set 3-13
- communications data set 3-14
- HELP library 3-21
- JCLLIB data set 3-23
- Load library 3-13
- log data set(s) 3-17
- log history/archive data sets 3-25

Installation requirements (*continued*)

file requirements (*continued*)

- Model CA-Netman Transaction File 3-23
- Options data set 3-22
- override library 3-22
- queue dump data set 3-18
- SASSCDSI SYSIN data set 3-15
- Statistics data set 3-20
- submit data set(s)/internal reader(s) 3-16
- VRM Dump data set 3-19
- workload planning data set(s) 3-20

index data set 3-7

JCL data sets 3-9

Job data set 3-6

job entry systems supported 3-2

memory 3-2

operating systems supported 3-2

queue files

- DASD restrictions 3-13
- organization 3-11
- space requirements 3-12
- status queues 3-10
- work queues 3-11

VRM data sets 3-8

workload balancing (WLB)

- implementing 3-42
- production environment, defining 3-38
- selection parameters, defining 3-44
- tape drive types, defining 3-53
- WLB macros, coding 3-55

Interfaces

- disaster recovery 7-35
- to external security packages 1-10
- TSO/ISPF
 - and virtual terminals 5-31

Internal

- events 2-13
- reader 2-14
- terminals 2-20

Internet, CA site 11-5

ISPF interface 5-31

J

JCL

- Attach Verification exit 9-21
- batch-only 6-13
- data sets 2-10, 3-9, 5-58
- Database Backup/Reload 7-6
- for formatting the communications data set 3-14
- ICOM execution 6-26
- log
 - archive 8-6

JCL (*continued*)

log (*continued*)

- dump 8-3
- purge 8-12
- online execution 6-5
- overrides 1-11
- recovery aid 7-42
- sample initialization file edit 5-86
- statement 5-57
- Submission exit 9-9

JCLCHECK statement 5-61

JCLLIB data set

- description 2-17
- space requirements 3-23

Job

- class structure 3-41
- Data Verification exit 9-19
- entry systems 3-2
- flow control
 - See* Work flow control
- N020 8-2
- N220 6-11
- Name Verification exit 9-8
- restart 1-10, 5-47
- start times 3-41
- statement 3-2

Job data set

- description 2-8
- space requirements 3-6

Job-program cross reference 9-41

K

Keywords, initialization file 5-4

L

L233COLD member 6-42

L2EXTT macro 3-73

L2XDSN macro 3-79

Line groups 5-28

LINE statement 5-31, 6-11

Load library

- placement 10-12
- space requirements 3-13

Log

- See also* Log data
- See also* Log data sets
- archives file 8-4, 8-9
- history file 8-4, 8-9
- usage 6-16

- Log data
 - backup (SASSHIS5) 7-38
 - input to SASSHIS5 8-4
 - preserving 7-38
 - recovery reports 7-38
- Log data set management
 - Archives Purge report 8-13
 - description 8-2
 - log dump JCL examples 8-3
 - log purge JCL example 8-12
 - SASSHIS5 (History Management program) 8-4
 - SASSHIS6 (Archives Purge program) 8-9
- Log data sets
 - description 2-16
 - file requirements 3-17
 - primary 5-45, 5-46, 8-2
 - secondary 5-45, 5-46, 8-2
 - space requirements 3-17
- Log History/Archive data sets
 - overview 2-16
 - requirements 3-25
 - space requirements 3-25
- Logical terminals 2-18
- Logoff exit (SASSXXFF) 9-30
- Logon
 - exit (SASSXXLX) 9-29
 - screen messages 5-12
- LOGON command 7-41
- LWLB command 3-69

M

- Magnetic tape, recovery 7-36
- Maintenance procedures 4-1
- Management level reporting 1-13
- Managing log data sets 8-2
- Masking 9-55
- Master station 2-18
- Master terminal
 - and TERM statement 5-35
 - defining 2-18
 - STATIONS statement 5-40
- Member \$\$INDEX 9-2
- Memory-resident queues 5-67, 5-68, 10-7
- Message Level Suppression 9-52
- Messages
 - broadcasting 5-12
 - format blocks, defining 5-24
 - input/output 5-24
- Migrations, queue files A-2
- MIM (ENQ) 10-12

- Model CA-Netman transaction file
 - overview 2-17
 - space requirements 3-23
- Modifications (USERMODs) 9-3
- MODIFY command
 - using instead of WTOR 6-30
- Modules
 - CARAGENB 9-53
 - SASSXU83 9-56
- Monitoring CA-7 performance 10-2
- Move Queue exit 9-32
- MOVQ start (Move Queues Restart) 6-21

N

- NCF system
 - overview 2-7
 - recovery 7-36
- NETMAN statement 5-71
- Network
 - overview 2-12
 - specifying on RESIDENT statement 5-10
- NEWS statement 5-12
- Normal reinitialization 6-39
- NOSCHDY parameter 3-33

O

- Offline verification of initialization file 5-86
- Online
 - access 2-18
 - assistance 1-7
 - CA-7 execution 6-2
 - exits
 - descriptions 9-8
 - register conventions 9-6
 - file requirements 3-13
 - utilities 1-10
- Operating Systems
 - and CA-7 3-2
 - considerations
 - dispatching priority 10-9
 - nonswappable 10-9
- Options data set
 - description 2-17
 - space requirements 3-22
- OPTIONS statement 5-72
- OS CANCEL command 6-25
- Override library
 - defined 2-17
 - space requirements 3-22

P

- PA key
 - assignment 1-7, 5-82, 5-83
- PAnn statements 5-83
- PARM values, CA-7 6-3
- Pattern masking 9-55
- Patterns 9-54
- Performance and tuning 10-1
- Performance reports 10-2, 10-3
- Personal Scheduling Verification exit 9-24
- PF key
 - assignment 1-7, 5-82, 5-83
- PFnn statements 5-83
- PFTERM statement 5-82
- Phone Services Directory 11-7
- Postinstallation testing 4-2
- Postprocess queue
 - overview 2-11
 - space requirements 3-10
- Preprocess queue
 - overview 2-11, 2-12
 - space requirements 3-10
- Prior-run queue
 - overview 2-11
 - space requirements 3-10
- Processing options
 - DBASE statement 5-43
 - OPTIONS statement 5-72
 - VRMOPTS statement 5-81
- Production control 1-6
- Productivity 1-13
- Program-DSNAME cross reference 9-42
- Program-job cross reference 9-41
- Purging log records 8-9

Q

- Queue
 - See also* Queue files
 - data sets
 - DASD device 5-69
 - placement of 10-12
 - Dump data set
 - and CA-7 DD statements 6-7
 - description 2-16
 - space requirements 3-18
 - Entry JCL exit 9-14
- Queue access method 3-11
- Queue files
 - active queue 2-11, 3-10
 - DASD restrictions 3-13

Queue files (*continued*)

- DASD-resident 5-67
- disk queue table (DQT) 2-11, 3-11
- identifying (FORMAT statement) 5-63
- memory-resident 5-67, 6-2, 10-7
- migrating A-2
- organization 3-11
- placement on
 - a single pack 10-10
 - fixed-head disk drives 10-11
 - multiple packs 10-10
- postprocess queue 2-11, 3-10
- preprocess queue 2-11, 3-10
- prior-run queue 2-11, 3-10
- ready queue 2-11, 3-10
- request queue 2-11, 3-10
- scratch queue 2-11, 3-11
- space requirements 3-12
- trailer queue 2-11, 3-11

Queue IOB count 10-5

R

- Ready queue
 - overview 2-11
 - space requirements 3-10
- Records
 - queue 9-31, 9-32
 - SMF purge 4-9
 - SMF type 14/15 9-54
- Recovery aid facility
 - log data reports 7-38
 - procedures 7-40
 - queue milestones 7-37
 - recovery aid JCL 7-42
 - reports, requesting 7-39
- REGION size
 - and PARM values 6-3
 - and trailer queue usage 6-2
 - CA-7 3-2, 6-2
- Register conventions 9-6
- Reinitializing using CAIRIM 6-39
- Release enhancements summary 1-1
- RELINK command and user exits 9-7
- Reload VRM 7-30
- Reporting
 - Automated Performance Analysis 1-13, 9-44, 10-2
 - CA-Earl reporting 1-13, 4-7
 - CA-Easytrieve Plus reporting 1-13
 - database cross-reference 9-40
 - heading specification 5-12
 - history 1-13, 8-1

- Reporting (*continued*)
 - installation verification 4-4
 - log 8-1
 - productivity 1-13
 - recovery 7-20
- Reports
 - performance 10-3
 - SASSBK00-1 7-20
 - SASSBK07-01 7-22
 - SASSHIS5 8-7
 - SASSHIS6 8-13
 - SASSHIS8 7-39
 - SASSRA01, Last Logged Status of Jobs report 7-38
 - SASSRA02, Generated Batch Terminal Interface Com-
mands report 7-38
 - SASSXREF-01 9-40
 - SASSXREF-02 9-40
 - U7TESTSV 4-4
- Request queue
 - definition 3-10
 - figure 2-12
 - purpose 2-11
 - tuning 10-4
- RESCHNG command 3-69
- Reserve 10-12
- Reserved DDname Table 9-53
- RESIDENT statement
 - and batch execution 6-11
 - defined 5-10
- RESTART statement 5-47
- Restricting access 1-10
- Running CAIRIM 6-38

S

- SASSBCLP (Batch Card Load program) 7-15
 - and DBPARMS deck 7-15
- SASSBCX1 (Batch Card Load program (BCLP) exit) 9-33
- SASSBK00 (Backup Database)
 - and backup 7-6
 - and recovery 7-13
 - overview 7-2
 - PARM values 7-9
 - using PARM data 9-38
- SASSCDSI
 - program 7-15
 - SYSIN data set 3-15
- SASSEXTL tables 3-70
- SASSEXTT tables 3-70
- SASSFMTA module 5-24
- SASSFMTH module 5-24

- SASSGPHJ module 9-48
- SASSHIS5 (History Management program)
 - backing up log data 7-38
 - description 8-4
 - file descriptions 8-6
 - flowchart 8-5
 - report descriptions 8-7
- SASSHIS6 (Archives Purge program)
 - description 8-9
 - file descriptions 8-12
 - flowchart 8-11
 - report descriptions 8-13
- SASSHIS8 (history reporting facility) 7-39
- SASSHR50 (print recovery aid reports) 7-39
- SASSICOM EXEC statement 6-27
- SASSMSGs module 9-52
- SASSPMDD table 9-53
- SASSPROG module 5-22
- SASSRA02, Generated Batch Terminal Interface Commands
report 7-38
- SASSRA03, Simulated LQ Display of Jobs report 7-38
- SASSSCJL module 9-14
- SASSSFCW module 9-17
- SASSSM5C module 9-13
- SASSSSM0 module 9-9
- SASSUTBL module 3-40, 3-53, 9-51
- SASSXDSN table 3-77
- SASSXREF-01 Report 9-40
- SASSXREF-02 Report 9-40
- SASSXU83
 - module 9-56
 - table 9-54
- SASSXX01 (Job Name Verification exit) 9-8
- SASSXX02 (JCL Submission exit) 9-9
- SASSXX03 (Utility Function Security Checking exit) 9-11
- SASSXX04 (ENQ/RESERVE exit) 9-13
- SASSXX05 (Queue Entry JCL exit) 9-14
- SASSXX07 (External DSN Security exit) 9-15
- SASSXX08 (Forecast Worksheet exit) 9-17
- SASSXX09 (Command exit) 9-18
- SASSXX10 (Job Data Verification) 9-19
- SASSXX11 (JCL Attach Verification) 9-21
- SASSXX12 (Database Load Processing Exit) 9-35
- SASSXX13 (SMF Feedback Exit) 9-22
- SASSXX14 (Personal Scheduling Verification Exit) 9-24
- SASSXX15 (CA-7 Console Terminal Output Exit) 9-26
- SASSXX16 (SMF WTO Message Exit) 9-27
- SASSXX17 (Browse Message Exit) 9-28
- SASSXXFF (Logoff exit) 9-30
- SASSXXLX (Logon/Password Verification exit) 9-29

- SASSXXQB (Move Queue exit (MOVQ)) 9-32
- SASSXXQD (Dump Queue exit (DMPQ)) 9-31
- SCALyyxx parameter 3-32
- SCHDAYS parameter 3-33
- SCHDYONLY parameter 3-33
- Schedule scan
 - and initialization 5-16
 - description 5-50
 - overview 2-4
 - parameters and performance 10-4
- SCHEDULE statement
 - and batch execution 6-12
 - coding 5-50
- SCNINCR parameter 5-50
- SCNSPAN parameter 5-50
- Scratch queue
 - overview 2-11
 - space allocations 3-11
- Security 1-10
 - DSN, external 9-15
 - overview 1-10
 - statement
 - and batch execution 6-12
 - parameters 5-39
- SHUTDOWN command
 - and TERM statement A-2
 - migrating queues 5-35
- Shutdown procedures 6-25
- SIMLOGON (VTAM) 5-37
- Single graphs 9-48
- SMF
 - Feedback exit 9-22
 - feedback processing 2-4
 - monitoring record types 2-6
 - purge records 4-9
 - type 14/15 records 9-54, 10-4
 - WTO Message exit 9-27
- Standard exit descriptions 9-33
- STARTIME (WLB macro) 3-43, 3-66
- Startup options
 - COLD start 6-20
 - DORM start 6-22
 - EDIT start 6-22
 - ERST start 6-19
 - FORM start 6-22
 - MOVQ start 6-21
 - overview 6-18
 - WARM start 6-18
- Startup procedures 6-17
- STATIONS statement 5-40, 6-12

- Statistics data set
 - description 2-16
 - space requirements 3-20
- Status queues 2-11
- STNCAL statement 5-41
- Submit data sets
 - and CPU statement 5-54
 - defined 2-16
 - internal reader(s) file requirements 3-16
 - uses 2-4
- Summary of enhancements 1-1
- Support file requirements 3-25
- Support Phone Services Directory 11-7
- Suppressing messages 9-52
- SVC
 - and SMF exits 2-6
 - Type IV 2-6, 3-2, 5-17
- SVCNO statement 5-17
- Symbolic index 5-58
- Syntax rules for initialization file statements 5-2
- SYSGEN macros 6-4
- System Management Facility
 - See* SMF
- System structure 2-2

T

- Tables
 - Batch Interface message table 9-49
 - Generic Unit Name/Device Type Table 9-51
 - graph 9-43
 - Reserved DDname Table 9-53
 - SASSEXTL 3-70
 - SASSEXTT 3-70
 - SMF Type 14/15 Record Exclusion/Inclusion Table 9-54
- Tape
 - drive usage 3-45
 - drives 3-40
- TAPE1 (WLB macro) 3-43, 3-58
- TAPE2 (WLB macro) 3-43, 3-60
- Tasks, tracking external 3-70
- Technical Support Phone Services Directory 11-7
- TERM statement 5-33, 6-11
- Terminal communications 2-18
 - batch access 2-19
 - batch terminal interface 2-19
 - BSAM terminal (browse data set) 2-20
 - logical terminals 2-18
 - online access 2-18
- Terminals
 - 3270-type access 5-27
 - assigning PF/PA keys 5-82, 5-83

- Terminals (*continued*)
 - CCI 2-20
 - dispatching priority 10-6
 - internal 2-20
 - line groups, defining 5-28
 - recovery 7-33
 - types, defining 5-33
 - workstations/logical, defining 5-40
- Test copy 4-8
- Testing
 - postinstallation 4-2
 - user exits 9-7
- Text editor 1-13
- Threshold priorities 3-42
- TNAME parameter 5-31
- Tracking external data sets 3-77
- Tracking external tasks 3-70
- Trailer queue
 - and memory 10-7
 - and region size 6-2
 - overview 2-11
 - space requirements 3-11
- Trailer step facility 2-19
- Troubleshooting 11-1, 11-10
- TSO/ISPF and virtual terminals 5-31
- Tuning CA-7 10-1
- TYPE
 - 14/15 SMF records 10-4
 - startup parameter 6-3

U

- U7SVC facility
 - figure 2-19
- UCC7ACTQ ddname 5-64
- UCC7CMDS 10-12
- UCC7DBASE statements 7-16
- UCC7DBVR (Database Verification program) 7-15
- UCC7DQTQ ddname 5-64
- UCC7NODE table 6-39
- UCC7PREQ ddname 5-64
- UCC7PRNQ ddname 5-64
- UCC7PSTQ ddname 5-64
- UCC7RDFL module 3-68
- UCC7RDYQ ddname 5-64
- UCC7REQQ ddname 5-64
- UCC7SCRQ ddname 5-64
- UCC7TRLQ ddname 5-64
- UCC7VTAM statement 5-27
- UID security 3-9
- UNIX System Services interface 3-81

- User exits
 - Batch Card Load program (SASSBCX1) 9-33
 - Browse Message Exit (SASSXX17) 9-28
 - CA-7 Console Terminal Output Exit (SASSXX15) 9-26
 - coding 9-3
 - Command exit (SASSXX09) 9-18
 - Database
 - Backup 9-38
 - Cross-reference 9-40
 - Reload 9-40
 - Database Load Processing Exit (SASSXX12) 9-35
 - Dump Queue (SASSXXQD) 9-31
 - ENQ/RESERVE (SASSXX04) 9-13
 - External DSN Access (SASSXX07) 9-15
 - Forecast Worksheet (SASSXX08) 9-17
 - JCL
 - Attach exit example 9-47
 - Attach Verification (SASSXX11) 9-21
 - Queue Entry (SASSXX05) 9-14
 - Submission (SASSXX02) 9-9
 - Job Data Verification (SASSXX10) 9-19
 - Job Name Verification (SASSXX01) 9-8
 - Logoff (SASSXXFF) 9-30
 - Logon/Password Verification (SASSXXLX) 9-29
 - Move Queue (SASSXXQB) 9-32
 - Personal Scheduling Verification Exit (SASSXX14) 9-24
 - SMF Feedback Exit (SASSXX13) 9-22
 - SMF WTO Message Exit (SASSXX16) 9-27
 - testing 9-7
 - USERMODs 9-3
 - Utility Function Security Checking (SASSXX03) 9-11
- User modifications
 - description 9-49
 - Generic Name/Device Type Table 9-51
 - graph definitions, changing
 - DEFGRPH macro 9-44
 - Graph Table definition 9-43
 - Message Level Suppression 9-52
 - reports, creating new
 - defining single graphs 9-48
 - Reserved DDname Table 9-53
- User table reload reinitialization 6-40
- User table reload reinitialization procedures 6-42
- USERMODs 9-2
- Utility commands 1-10
- Utility Function Security Checking exit 9-11

V

- Verifying
 - initialization file offline 5-86

Verifying the initialization file offline 5-86

Virtual

queues 5-67, 5-68, 10-7

terminals

and LINE statement 5-31

and TERM statement 5-33

and UCC7VTAM statement 5-27

VRM (Virtual Resource Management) 1-8

and COLD or FORM starts 6-20, 6-22

and recovery 7-28

backup 7-28

CA7VBK program 7-28

CA7VRL program 7-28

data set

overview 2-8

space requirements 3-8

device definitions 5-79

dump data set

file requirements 3-19

overview 2-16

space requirements 3-19

reload 7-30

support 5-45

VRM device control 3-38

VRMOPTS statement 5-81

VTAM

ACB 5-27

options, defining 5-27

terminals

defining 5-27

recovery 7-33

W

WARM start (Normal Restart) 6-18

WEEKDAYS parameter 3-34

WEEKENDS parameter 3-34

Wildcards 9-55

WLB

DFLTWLB keyword 5-74

implementing 3-42

macros, coding

CLBARR 3-65

CPU 3-62

INTR 3-63

rules 3-55

STARTIME 3-66

TAPE1 3-58

TAPE2 3-60

WLBPDEF 3-57

WLBPEND 3-67

module

generation 3-68

WLB (continued)

module (continued)

scheduling 3-69

production environment, defining

CPU use 3-41

initiators and job class structure 3-41

job priority threshold 3-42

job start times 3-41

overview 3-39

tape drives 3-40

questionnaire 3-39

selection parameters, defining 3-44

tape drive types, defining 3-53

WLBPDEF (WLB macro) 3-43, 3-57

WLBPEND (WLB macro) 3-43, 3-67

WLP

backup and recovery considerations 7-35

data sets 2-16, 3-20

space requirements 3-20

Work flow control 1-8

Work flow through the queues 2-12

Work queues 2-11

Workload

backup and recovery considerations 7-35

balancing

See also WLB

activating/deactivating 3-69

defining selection parameters 3-44

overview 1-12, 3-38

questionnaire 3-45

documentation 1-12, 7-36

forecasting 1-11

planning 1-12

scheduling 1-8

sequencing 1-8

Workload balancing

and VRM device control 3-38

Workstation

calendar, STNCAL statement 5-41

network control

See Work flow control

X

XPS MONITOR 5-18

XPS ROUTER 5-18

XPS server support 5-29, 5-34

XWLB command 3-69

